AD-A074 438 NAVAL POSTGRADUATE SCHOOL MONTEREY CA F/G 12/1

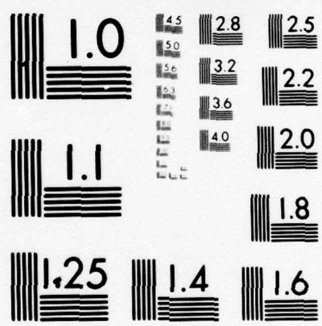
NUMERICAL INVERSION OF LAPLACE TRANSFORMS.(U)

JUN 79 J H DUNCAN

UNCLASSIFIED NL

1 OF 2

ADA 074438




MICROCOPY RESOLUTION TEST CHART

# LEVEL II

B.S.

2

AD A074438

# NAVAL POSTGRADUATE SCHOOL
## Monterey, California

9 Master's thesis,

# THESIS

12 191p.

DDC FILE COPY

6 NUMERICAL INVERSION OF LAPLACE TRANSFORMS.

by

10 John Hollis Duncan

11 June [redacted]79

Thesis Advisor: J. E. Brock



Approved for public release; distribution unlimited.

JOB

251 450

79 09 24 085

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

| REPORT DOCUMENTATION PAGE | | READ INSTRUCTIONS BEFORE COMPLETING FORM |
|---|---|---|
| 1. REPORT NUMBER | 2. GOVT ACCESSION NO. | 3. RECIPIENT'S CATALOG NUMBER |
| 4. TITLE (and Subtitle)<br>Numerical Inversion of Laplace Transforms | | 5. TYPE OF REPORT & PERIOD COVERED<br>Master's Thesis; June 1979 |
| | | 6. PERFORMING ORG. REPORT NUMBER |
| 7. AUTHOR(s)<br>John Hollis Duncan | | 8. CONTRACT OR GRANT NUMBER(s) |
| 9. PERFORMING ORGANIZATION NAME AND ADDRESS<br>Naval Postgraduate School<br>Monterey, California 93940 | | 10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS |
| 11. CONTROLLING OFFICE NAME AND ADDRESS<br>Naval Postgraduate School<br>Monterey, California 93940 | | 12. REPORT DATE<br>June 1979 |
| | | 13. NUMBER OF PAGES |
| 14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office) | | 15. SECURITY CLASS. (of this report)<br>Unclassified |
| | | 15a. DECLASSIFICATION/DOWNGRADING SCHEDULE |

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Laplace Transformation
Integral Transformations
Inversion of Laplace Transformation
Numerical Inversion
Numerical Integration
Inversion Integrals

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

Inversion of Laplace transforms has been accomplished by a numerical integration along appropriate paths in the complex plane. Two general procedures have been used. → next page

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE
S/N 0102-014-6601 |
(Page 1)

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

The simpler and more economical employs a simple path, such as a parabola, which bends to the left. Accuracy is maintained by monitoring the oscillation of the integrand. A second method employs a steepest descent contour.

| Accession For | |
|---|---|
| NTIS GRA&I | ☒ |
| DDC TAB | ☐ |
| Unannounced | ☐ |
| Justification | |
| By | |
| Distribution/ | |
| Availability Codes | |
| Dist. | Avail and/or special |
| A | 23 CP |

Approved for public release; distribution unlimited.

# Numerical Inversion of Laplace Transforms

by

John Hollis Duncan
Lieutenant Commander, United States Navy
B.S., University of South Carolina, 1967

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN MECHANICAL ENGINEERING

from the

NAVAL POSTGRADUATE SCHOOL
June 1979



Author: John Hollis Duncan

Approved by: John E. Brock
Thesis Advisor

R. E. Newton
Second Reader

P. J. Marto
Chairman, Department of Mechanical Engineering

William M. Tolles
Dean of Science and Engineering

## ABSTRACT

Inversion of Laplace transforms has been accomplished by a numerical integration along appropriate paths in the complex plane. Two general procedures have been used. The simpler and more economical employs a simple path, such as a parabola, which bends to the left. Accuracy is maintained by monitoring the oscillation of the integrand. A second method employs a steepest descent contour.

TABLE OF CONTENTS

## LIST OF ILLUSTRATIONS

ACKNOWLEDGEMENT

The author would like to take this opportunity to thank Professor J. E. Brock, Department of Mechanical Engineering, Naval Postgraduate School, for the help and guidance he professionally provided during the accomplishment of this work. The insight he contributed and his genuine interest and enthusiasm for resolution of the problem were superb.

The author also wishes to express his appreciation to Professor R. E. Newton and Associate Professor M. D. Kelleher, Department of Mechanical Engineering, Naval Postgraduate School, for their contributions to the author's general education as well as their guidance and response to questions on the enclosed work.

Lastly, the author would like to express appreciation to my wife and children who sacrificed so much during this tour of duty and provided the inspiration necessary to complete the project.

# I. INTRODUCTION

If F(t) is an integrable function of the real variable t, the function f(s) defined by the integral

$$f(s) = \int_0^\infty e^{-st}\, F(t)dt \tag{1}$$

is called the Laplace transformation, or the Laplace transform, of F(t) and is frequently indicated by the notation

$$f(s) = \mathcal{L}[F(t)] \tag{2}$$

The Laplace transformation is a linear integral transformation which is widely used in applied mathematics and technology. A typical application is as follows. An unknown function F(t) satisfies a certain differential equation and specified end conditions. Employing theorems applicable to the Laplace transform, a function f(s) is determined as the solution of an algebraic equation which corresponds, in an appropriate fashion, to the differential equation and end conditions of F(t). Then the problem is reduced to the following task: having established f(s), it is necessary to determine the function F(t) for which $f(s) = \mathcal{L}[F(t)]$.

This problem is called finding the inverse (Laplace) transform of the function f(s). This is customarily written as

$$F(t) = \mathcal{L}^{-1}[f(s)] \tag{3}$$

It may be shown that the following inversion integral

$$F(t) = \frac{1}{2\pi i} \int_C e^{st} f(s)ds \tag{4}$$

accomplishes the desired inversion. In this expression s is the complex variable

$$s = x + iy \tag{5}$$

and the integration is along a suitable path in the complex s-plane.

If in fact there is a function F(t) such that the given $f(s) = \mathcal{L}[F(t)]$, then the function determined by equation (4) is, indeed, the function F(t). The result is unique in the following sense. If

$$f(s) = \int_0^\infty e^{-st} F_1(t)dt \tag{6}$$

and

$$F_2(t) = \frac{1}{2\pi i} \int_C e^{st} f(s)ds \tag{7}$$

then $F_1(t)$ and $F_2(t)$ differ only on a set of Lebesgue measure zero.

In this thesis numerical methods will be utilized to perform the integration of equation (4) and functions for which the inversion is successful will correspond to functions F(t) which are at least piecewise continuous, having only isolated discontinuities, if any are present at all. If F(t) is discontinuous for t = a, then the recovered function satisfies the condition

$$F(a) = \frac{1}{2} \lim_{\varepsilon \to 0} [F(a+\varepsilon) + F(a-\varepsilon)] \tag{8}$$

The path of integration, indicated by C in equation (4) is frequently described as an infinitely long, vertical straight line x = constant, to the right of any singularities of f(s). (A theorem of complex variables shows that f(s) is analytic in its region of convergence, a right-hand plane, except for isolated singularities.) This straight vertical contour is frequently called the Bromwich contour.

Usually inversion of a Laplace transform, that is finding $F(t) = \mathcal{L}^{-1}[f(s)]$, is accomplished by use of a table of transform pairs. One of the most complete of these is that of Roberts and Kaufman [14]. Other such tables are listed as references [5], [6], [7], and [13].

Several theorems concerning the Laplace transform extend the usefulness of such transformation pair tables. For example

$$\mathcal{L}^{-1}[f_1(s) + f_2(s)] = \mathcal{L}^{-1}[f_1(s)] + \mathcal{L}^{-1}[f_2(s)] \tag{9}$$

However, it is not an infrequent occurrence that one is called upon to find F(t) in a case where the tables and the theorems are of no direct help. There has developed a considerable literature concerning this problem.

One obvious method is to attempt to expand the given f(s) as a series of functions $f_i(s)$ for which the inverses are known. Then using the linear property indicated in equation (9), the result may be expressed as

$$\mathcal{L}^{-1}[f(s)] = \mathcal{L}^{-1}[\Sigma a_i f_i(s)] = \Sigma a_i \mathcal{L}^{-1}[f_i(s)] \tag{10}$$

The integration indicated in equation (4) can also be performed by various analytical and approximate methods. In particular, one can perform the integration along the Bromwich contour by a numerical procedure. Much of the literature is devoted to such methods.

In general these methods amount to the following. A finite number of points along the Bromwich contour are determined according to some law and the integrand is evaluated at such points. There are associated weight factors and the integral is approximated as a sum of products of integrand values times weight factors. One of the most widely used of such methods is described by Salzer [15], [16], and [17]. Other procedures of the type are treated in [18], [20], [2], [8], [9], [10], and [21].

It is interesting to note that the same classical polynomials which are useful in the expansions indicated by equation (10) are also encountered in locating points for evaluation using a numerical integration scheme so that the final algorithm may be the same even though the original motivation was quite different.

Other methods which do not fall clearly into the classes of methods discussed above have been presented by Widder [22], and Bellman, Kalaba, and Lockett [3].

These approximate methods have not always been successful. Hiep [12] employed Salzer's method for numerical inversion on a problem of heat transfer in porous media and found it difficult to obtain results which did not exhibit physically incorrect behavior, e.g., in the vicinity of a point of engineering interest, the temperature of the medium fell below sink temperature and rose above source temperature. In similar work on a conjugated heat transfer problem Zargary [23] encountered the same difficulty, and, after devoting considerable time and attention to the problem, was forced to abandon the use of numerical inversion.

To overcome such difficulties in problems of these kinds, we have reverted to direct numerical integration of equation (4) along appropriate contours in the s-plane. Our procedures, which appear thus far to be efficient and reliable, are described in what follows.

# II. DISTORTED CONTOURS

We have found that great advantage accrues from distorting the Bromwich contour in ways which will subsequently be described.

According to theorems concerning analytic functions of a complex variable, the Bromwich contour can be distorted arbitrarily as long as the process of distortion does not cause the contour C to cross over a singularity of f(s) and as long as the two ends of C extend to infinity in a manner which preserves the convergence of the integral in equation (4).

Thus, with the supposition that f(s) is analytic, with the exception of isolated singularities, and is real valued for real s, let g(s) represent the integrand

$$g(s) = e^{st} f(s) = u(s) + iv(s) \tag{11}$$

where u(s) and v(s) are the real and imaginary harmonic components of g(s). Furthermore, assume that the contour C consists of the two symmetrical parts $C^*$ and $C^{**}$ as shown in Figure 1. Consideration of the contributions to $\mathcal{L}^{-1}[f(s)]$ over arcs $ds_1$ on $C^*$ and $ds_2$ on $C^{**}$ yields the following:

$$\begin{aligned} dx_2 &= -dx_1 \\ dy_2 &= dy_1 \\ u_2 &= u_1 \\ v_2 &= -v_1 \end{aligned} \tag{12}$$

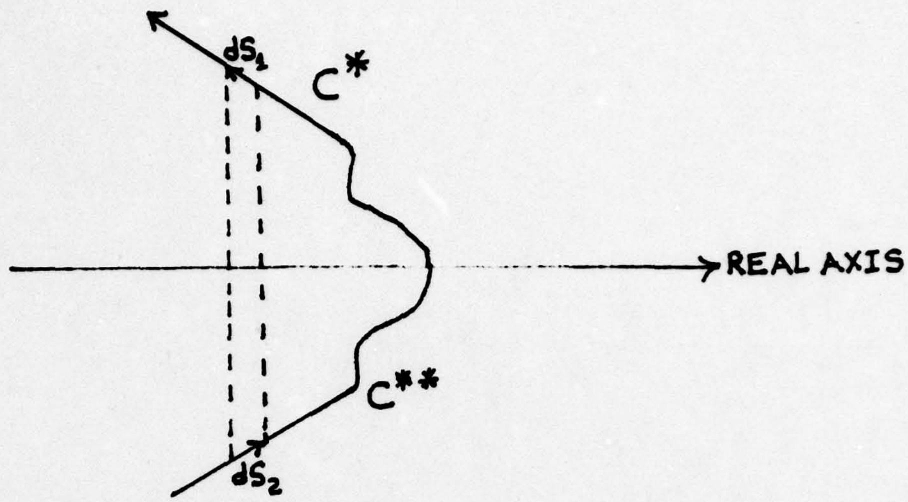



FIGURE 1. Distorted Bromwich Contour. Contour is distorted into $C^*$ and its reflection $C^{**}$.

Thus,

$$\frac{1}{2\pi i}[g_1 ds_1 + g_2 ds_2] = \frac{1}{\pi}[v_1 dx_1 + u_1 dy_1] \tag{13}$$

and, therefore,

$$F(t) = \frac{1}{\pi} \int_{C^*} [vdx + udy] \tag{14}$$

Equation (14) is, of course, further simplified if $C^*$ is selected to be a path along which v=0. Additional theoretical reasons exist for such a selection. Such a path will subsequently be referred to as a path of steepest descent. Numerical integration along such a path, or a close facsimile thereof, was done by Esch [11] in 1957 and Carrier, Krook, and Pearson [4] recommended further exploration of this procedure. However, no further such investigation seems to have been done until the present thesis. In Chapter V, a description of an algorithm for performing numerical integration along a path $C^*$, for which v=0 over a substantial portion of its range, will be presented.

Almost by accidental discovery, the present work has revealed that it is computationally more efficient simply to make an arbitrary choice for the contour $C^*$ which satisfies the following criteria:

(1) $C^*$ is a smooth continuous curve of the form

$$s(p) = x(p) + iy(p) \tag{15}$$

where p is a real parameter $0 \leq p \leq \infty$.

(2) $C^*$ lies to the right of all singularities of f(s).

(3) $C^*$ does not extend sufficiently far to the right to cause computational difficulties due to the factor $e^{st}$.

(4) $C^*$ approaches infinity in such a way that x approaches $-\infty$ and y approaches $\infty$.

(5) Oscillation of the integrand is not excessive. (Otherwise there may be loss of accuracy due to the sampling rate of the integration algorithm seriously mismatching variations in the integrand and/or positive and negative contributions nearly cancelling each other.)

Such a contour will be referred to as a simple parameterized contour. Discussion of how to assure satisfying criteria

(2), (3), and (5) will be deferred until after the development of the actual algorithm.

The integral shown in equation (14) is approximated by the simple trapezoidal sum

$$F(t) = \frac{1}{2\pi} \sum_{k=1}^{M} \{[v(s_k)+v(s_{k+1})][x_{k+1} - x_k]$$

$$+ [u(s_k)+u(s_{k+1})][y_{k+1} - y_k]\} \tag{16}$$

where

$$s_k = x_k + iy_k = x(p_k) + iy(p_k) \tag{17}$$

The arbitrarily chosen functions x(p) and y(p) are monotonically increasing and $p_1, p_2, \ldots, p_n$ is an increasing sequence.

A very simple example, but one which has been employed very successfully, is given by the following equations

$$\begin{aligned} x &= A - Bp^2 \\ y &= p \\ p &= 0, \Delta, 2\Delta, \ldots \end{aligned} \tag{18}$$

The real number A is selected so as to satisfy criteria (2) and (3) above. The real number B has, usually, been assigned a value of one.

The sum includes only a finite number of terms. It is necessary to provide an appropriate criterion for terminating the process of summation. This will be discussed later.

In order to monitor the possibility of inaccurate evaluations resulting from oscillations, in addition to the sum F(t) given by equation (16), the sum G(t) given by

$$G(t) = \frac{1}{2\pi} \sum_{k=1}^{M} \left| \{[v(s_{k+1})-v(s_k)][x_{k+1}-x_k] + [u(s_{k+1})+u(s_k)][y_{k+1}-y_k]\} \right| \tag{19}$$

is also obtained. This is the sum of the absolute values of the addends to the sum (16). Additionally, the number of times successive addends are of opposite signs is recorded.

The present numerical experiments, to be described later, were generally so successful that the effect of oscillation could not be seen. However, in some cases where poor contours $C^*$ were deliberately chosen, it was found that inaccurate results were obtained if G(t) was many orders of magnitude (e.g., $10^6$) times as great as F(t).

## III. THE COMPUTER PROGRAM AND ITS PRINCIPAL SUBROUTINES

Previous discussion proposed two alternatives in terms of advantageous distortion of the Bromwich contour upon which to perform the numerical integration of equation (16). The simple parameterized curve has been experimentally found to be the more efficient of the two methods and its algorithm will be developed at this time. The steepest descent procedure will be treated in Chapter V.

The computer program and subprograms which implement the simple procedure developed in the preceding chapter are all written in the FORTRAN language, using double precision arithmetic. They have been tested and debugged on an IBM 360/67. For machines having a larger mantissa, single precision arithmetic might prove to be satisfactory.

SUBROUTINE VALUE is a user supplied routine which implements computation of the g(s) given by equation (11) for the desired transform f(s). This subroutine calculates the u(s) and v(s) corresponding to a given x and y.

Either complex or real arithmetic may be utilized in this program, although complex arithmetic offers a decided advantage in convenience and simplicity. However, the user should be aware of the potential dangers in unintentional exchange of sheets of a Riemann surface when investigating a multi-valued function. More will be said about this later.

Also, although FORTRAN manuals state or imply that the FUNCTIONS DREAL and DIMAG are implemented in all compilers, this may not actually be the case. Accordingly it is prudent to add the following statements after SUBROUTINE VALUE.

```
FUNCTION DREAL(CPLX)
REAL*8 CPLX(2)
DREAL = CPLX(1)
RETURN
END
FUNCTION DIMAG(CPLX)
REAL*8 CPLX(2)
DIMAG = CPLX(2)
RETURN
END
```

SUBROUTINE CURVE is also a user supplied routine which calculates the x and y values for each consecutive incrementation of the parameter p of equation (17), thus locating a point on the simple parameterized contour. Equation (18) provides an illustrative and highly effective example of parabolic form which has been employed almost exclusively during this investigative work.

SUBROUTINE INCREP formulates the increasing sequence of the parameter p as given by the following equation

$$p = p + \Delta \qquad (20)$$

Limited analysis with other than equally spaced incrementation of points along the contour of integration did not exhibit any enhancing capabilities and was not continued.

The main program accomplishes the integration of equation (14) using a trapezoidal summation as described by equation (16). In addition, the absolute value of the integrand is summed in accordance with equation (19). Also, the number of times a sign changes occurs between successive evaluations of the addend of equation (16) is recorded. These features afford the user an opportunity to monitor potential oscillation effects upon the accuracy of the results. Clock time and the number of calls to SUBROUTINE VALUE required for performing the numerical contour integration are maintained as measures of the computational efficiency of the numerical inversion procedure.

Termination of numerical integration occurs when each of N successive evaluations of the addend of equation (19) has magnitude less than a specified epsilon. We have usually used $N = 5$. This is a very stringent requirement and could probably be relaxed resulting in some saving of computational time and with but minor loss of accuracy.

In addition to the termination criterion described in the previous paragraph, other program inputs include the increment $\Delta$ of equation (18), appropriate numerical values of any constants of the function $f(s)$, the starting position A of equation (18) and the value of t.

The program output prints the starting position of the integration contour, the values given by equations (16) and (19), the final values of x and y at the termination of integration, the number of changes of sign between successive addends in equation (16), the total number of calls made to SUBROUTINE VALUE during the process, and the clock time required for numerical integration.

As an illustration consider the following example where

$$f(s) = \frac{s}{s^2 + \alpha^2} \tag{21}$$

A suitable form for SUBROUTINE VALUE is as follows:

```
      SUBROUTINE VALUE (X, Y, T, AL, U, V, MMMMM)
C     THIS IS TABLE ENTRY NUMBER 09 OF LAPLACE TRANSFORMS BY SPIEGEL
      IMPLICIT REAL*8 (A,B,D-H,O-Z)
      IMPLICIT COMPLEX*16 (C)
      MMMMM = MMMMM + 1
      ZERO=0.D+0
      CS=DCMPLX(X,Y)
      CAL=DCMPLX(AL,ZERO)
      CT=DCMPLX(T,ZERO)
      CDEN=CS**2+CAL**2
      C=CS/CDEN
      CEXP=CDEXP(CS*CT)
      C=C*CEXP
      U=DREAL(C)
      V=DIMAG(C)
      RETURN
      END
```

(Note: The parameter $\alpha$ is denoted by the FORTRAN variable AL.)

If the contour indicated by equation (18) were chosen and numerical inversion were performed upon the f(s) given by equation (21) for the case where $A = 0.3$, $\Delta = 0.1$, $\alpha = 0.125$, $N = 5$ and $\varepsilon = 1.D\text{-}11$, the evaluation of $F(2\pi)$ would yield the following output

| AA | $F(2\pi)$ | $\lvert G(2\pi)\rvert$ | $X_F$ | $Y_F$ | LOSC | MMMMM |
|---|---|---|---|---|---|---|
| 0.3 | 7.07107D-1 | 1.32635D+0 | -5.46 | 2.4 | 4 | 25 |

where AA, LOSC, and MMMMM are the FORTRAN program names for the contour starting position, the number of sign changes between successive addends, and the total number of calls made to SUBROUTINE VALUE, respectively. Similar results for the almost 100 functions f(s) tested during the course of this research are listed in Appendix B.

## IV. AUXILIARY SUBROUTINES

Chapter III briefly mentioned the potential dangers inherent in the utilization of complex arithmetic within SUBROUTINE VALUE when the function f(s) is multi-valued.

If f(s) has branch points or winding points it is important to assure that one remains on the same branch of the function as one proceeds along the path of integration. Available FORTRAN operations do not assure this and it is necessary to employ specially programmed algorithms. The present discussion is limited to the two simple cases, raising of a complex number to a non-integer power and taking the logarithm of a complex number.

The essence of the matter lies in the FORTRAN operation DATAN2(y,x), or its equivalents, which must be relied upon to provide the argument of a complex number in polar form. This operation always provides a result in the range $-\pi < \theta \leq \pi$, so that if Arg (z) actually passes through the values $(2n+1)\pi$, for integer n, the FORTRAN produced value of Arg(z) experiences a jump of $\pm 2\pi$. If one is determining the logarithm of z or a non-integer power of z, shifting of branch will take place unless the continuity of Arg(z) is restored.

SUBROUTINES CPOWER and CLOG provide for this continuity by monitoring changes in Arg(z) and adding or subtracting $2n\pi$, whenever it is appropriate to do so, to the value of

DATAN2(y,x). As they are presenting written, each provides for up to five separate calls, within SUBROUTINE VALUE, to either SUBROUTINE CPOWER or SUBROUTINE CLOG.

These subroutines are listed below,

```
      SUBROUTINE CPOWER (ZIN,ZOUT,P,J,NCALL)
      IMPLICIT REAL*8 (A-H,O-Z)
      COMPLEX*16 ZIN, ZOUT
      DIMENSION TOLD(5), NCALL(5), NPI(5)
      IF(NCALL(J).EQ.1) NPI(J)=0
      OH = 0.D+0
      X = DREAL(ZIN)
      Y = DIMAG(ZIN)
      R = DSQRT((X*X) + (Y*Y))
      TRONG = DATAN2(Y,X)
      IF(NCALL(J).EQ.1) GO TO 5
      PROD = TRONG*TOLD(J)
      IF(PROD.GT.OH.OR.X.GT.OH) GO TO 5
      IF(TRONG.LT.OH) NPI(J) = NPI(J) + 2
      IF(TRONG.GT.OH) NPI(J) = NPI(J) - 2
    5 RP = R**P
      EN = NPI(J)
      TRITE = TRONG + (EN*3.14159265358979324D+0)
      TP = TRITE*P
      X = RP*DCOS(TP)
      Y = RP*DSIN(TP)
      ZOUT = DCMPLX(X,Y)
      TOLD(J) = TRONG
      NCALL(J) = NCALL(J) + 1
      RETURN
      END
```

```
      SUBROUTINE CLOG (ZIN, ZOUT, J, MCALL)
      IMPLICIT REAL*8 (A-H,O-Z)
      COMPLEX*16 ZIN, ZOUT
      DIMENSION TOLD(5), MCALL(5), NPI(5)
      IF(MCALL(J).EQ.1) NPI(J) = 0
      OH = 0.D+0
      X = DREAL(ZIN)
      Y = DIMAG(ZIN)
      R = DSQRT((X*X) + (Y*Y))
      TRONG = DATAN2(Y,X)
      IF(MCALL(J).EQ.1) GO TO 5
      PROD = TRONG*TOLD(J)
      IF(PROD.GT.OH.OR.X.GT.OH) GO TO 5
      IF(TRONG.LT.OH) NPI(J) = NPI(J) + 2
      IF(TRONG.GT.OH) NPI(J) = NPI(J) - 2
    5 REAL = DLOG(R)
      EN = NPI(J)
      TRITE = TRONG + (EN*3.14159265358979324D+0)
      ZOUT = DCMPLX(REAL,TRITE)
      TOLD(J) = TRONG
      MCALL(J) = MCALL(J) + 1
      RETURN
      END
```

If either algorithm is to be utilized by SUBROUTINE VALUE, the first requirement is to zero the applicable integer array, NCALL or MCALL, within the main program. Then, in SUBROUTINE VALUE, the calls to SUBROUTINE CPOWER are of the form

```
CALL CPOWER(ZIN,ZOUT,P,J,NCALL)
```

where ZIN is the COMPLEX*16 operand, ZOUT is the COMPLEX*16 output, P is the REAL*8 power, and J is the INTEGER*4 index which tells whether this is the first, second,..., call to SUBROUTINE CPOWER within SUBROUTINE VALUE. The relationship obtained is

$$ZOUT = (ZIN)^{P} \tag{22}$$

SUBROUTINE CLOG is used similarly. Calls made from SUBROUTINE VALUE are of the form

```
CALL CLOG(ZIN,ZOUT,J,MCALL)
```

where

$$ZOUT = Ln(ZIN) \tag{23}$$

J plays the same role as in SUBROUTINE CPOWER. There is no P. It should be noted that the array MCALL replaces NCALL.

## V. THE STEEPEST DESCENT CONTOUR

The theoretical advantages of a steepest descent contour have been mentioned previously. Consider a contour in the complex plane along which v=constant. As shown in Figure 2, let the $\xi$ axis be tangent to the contour with the $\eta$ axis perpendicular to the $\xi$ axis in the same sense that y is perpendicular to x.

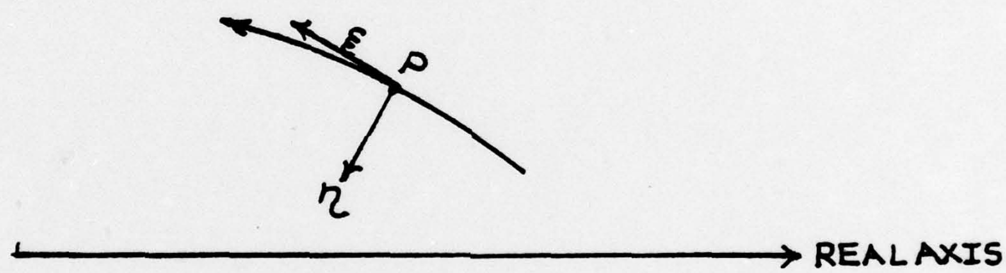



Figure 2. Steepest Descent Contour. Contour along which v is constant. Axis $\xi$ is tangent at point P. Axis $\eta$ is normal.

Recall equation (11),

$$g(s) = e^{st} f(s) = u(s) + iv(s) \tag{11}$$

By the Cauchy-Riemann relationships

$$\frac{\partial u}{\partial \xi} = \frac{\partial v}{\partial \eta} \; ; \; - \frac{\partial v}{\partial \xi} = \frac{\partial u}{\partial \eta} \tag{24}$$

Along a v = constant contour

$$\frac{\partial v}{\partial \xi} = 0 \; ; \; \frac{\partial u}{\partial \eta} = 0 \tag{25}$$

Such a $v = c$ = constant contour divides the plane, locally at least, into two regions: $R_1$ with $v > c$ and $R_2$ with $v < c$. Consequently along such a contour $\frac{\partial v}{\partial \eta}$ is one signed, and, therefore, $\frac{\partial u}{\partial \xi}$ is one signed. $u \to 0$ along the $v = c$ contour, in the direction of positive $\xi$. Therefore, there is no oscillation in u and, obviously also, there is none in v. Since $u \to 0$ and $v \to 0$ along a suitable integration contour, the selection of $c = 0$ assures that this results.

This second program to be discussed does partially follow a path of steepest descent. First it integrates vertically upward from a chosen point on the real axis (along a Bromwich contour) until intersection with a specified $v = 0$ contour occurs. Then it employs a tracking algorithm to obtain successive points on this contour along which it then integrates, moving to the left.

SUBROUTINE VALUE has the same role as that described in Chapter III, namely, it calculates the u(s) and v(s) in accordance with equation (11) corresponding to a given x and y. For the initial part of the algorithm, the integration along the Bromwich contour, the value of x is constant, and y is increased in increments by $\Delta y$. Numerical integration in accordance with equation (16) is conducted just as it was done in the previous case.

On the real axis, obviously, $v = 0$. With an initial incrementation in the vertical direction, v assumes a value other than zero. The sign of v is recorded and the incrementation process is continued until the sign of v changes. This constitutes a contour crossing. The procedure continues until a predetermined number of such crossings are obtained, and then a simple linear interpolation is performed to locate the specific ordinate value y for which $v = 0$. Usually one might wish to use the first $v = 0$ contour above the real axis, but the second, third, or other such contour may be selected.

The last $(y_i, v_i)$ values prior to the desired contour crossing and the first $(y_j, v_j)$ values after crossing are utilized in the linear interpolation process to predict YNEW, for which $v = 0$. SUBROUTINE VALUE provides the actual VNEW value corresponding to YNEW. Depending upon the sign of VNEW, the appropriate coordinate pair $(y_i, v_i)$ or $(y_j, v_j)$ is replaced by (YNEW,VNEW) and the interpolation is repeated with this refinement. Termination of the interpolation occurs when YNEW results in a magnitude of VNEW which is less in magnitude than a prescribed tolerance limit.

Once the values of x,y,u, and v have been established for the first point on the desired $v = 0$ contour in the previous process, the second, or tracking portion of the algorithm commences.

SUBROUTINE XMARCH is called to provide a new coordinate pair (x,y) in the negative sense of the real axis from the known, or old, coordinate pair $(x_o, y_o)$. This is accomplished by the following relationships

$$x = x_o - R\cos(\theta) \quad (26)$$

$$y = y_o + R\sin(\theta) \quad (27)$$

where (R,θ) are the polar coordinates of a search pattern measured from the known coordinates $(x_o,y_o)$ on the v = 0 contour. The angle θ is set at zero radians for the first call to SUBROUTINE XMARCH. SUBROUTINE VALUE is then called to provide the value of v at this point. The sign of v establishes upon which side of the v = 0 contour the new point lies.

SUBROUTINE ROTATE is then called to increment the angle theta in an appropriate sense. A sweeping search is then conducted by SUBROUTINES ROTATE, XMARCH, AND VALUE until a sign change is detected for v values between increments, thus, indicating a crossing of the v = 0 contour.

The last $(\theta_i,v_i)$ coordinate pair prior to a crossing and the first $(\theta_j,v_j)$ coordinate pair after a crossing are utilized in a linear interpolation scheme, analogous to that of the first program segment, to refine the location of the new point on the v = 0 contour.

Once the second point on the v = 0 contour is established, the tracking algorithm may be enhanced by predicting the third point on the contour to lie at the same angle θ from the second point. As before, the sign of v at this new location permits a rotational search to be conducted in the appropriate sense. For small values of the radius vector R (R=.025 is built into the program, but may be changed by the user), this approximation has been found, in general, to be a reasonable one.

The addends of equations (16) and 19) are evaluated at each step in the marching process within the main program and, once again, termination of the numerical integration occurs when each of N successive addend contributions is less than a prescribed epsilon.

Required inputs to the main program are the index specifying the desired $v = 0$ contour, the starting position on the real axis, the increment of the vertical march along the Bromwich contour, the termination criterion, any constants associated with the function $f(s)$, and the value of $t$. These inputs are represented by the program variables NREG, AA, DELY, EPS, (AL,BE,...), and T, respectively.

The simple parameterized contour offers one decided advantage over the steepest descent contour: each call made to SUBROUTINE VALUE is productive, in the former case, in the sense that it is utilized to compute a contribution of the addend of equation (16). The latter algorithm requires calls to SUBROUTINE VALUE to perform the linear interpolation procedures, which are non-productive in terms of addend computation. This represents increased computer time. The present investigation has shown that in each of the numerous cases which have been examined, the simple parabolic path indicated by equation (18) is more efficient and fully as accurate as a steepest descent contour.

## VI. SHANKS' ACCELERATOR

The effectiveness of a family of non-linear sequence to sequence transformations in accelerating the convergence of (some) slowly convergent sequences and in inducing convergence in (some) divergent sequences has been reported by Shanks [19]. If $\{A_n\}$ $(n=0,1,2,\ldots)$ is a sequence of numbers, we form a sequence of sequences $\{A_{k,n}\}$ which, for convenience, we write as $\{A(k,n)\}$. The integer k indicates the "order" of the sequence, with $A_n = A(0,n)$; the integer n indicates the position of the term in the sequence. The rule for forming the sequence $\{A(k+1,n)\}$ from the sequence $\{A(k,n)\}$ is

$$A(i,j) = 0 \quad \text{if} \quad j < 2i$$

$$A(k+1,n) = \frac{A(k,n)A(k,n-2) - A(k,n-1)^2}{A(k,n)+A(k,n-2)-2A(k,n-1)} \tag{28}$$

$$n=2k+3,2k+4,\ldots.$$

To illustrate the power of this computational device consider the application of equation (28) and its iterates to the first ten terms of the alternating series

$$\operatorname{Ln}(2) = 1 - \frac{1}{2} + \frac{1}{3} - \frac{1}{4} + \ldots \tag{29}$$

The results are

| n | A(0,n) | A(1,n) | A(2,n) | A(3,n) | A(4,n) |
|---|---|---|---|---|---|
| 1 | 1.000CC00000 | 0.0 | 0.0 | 0.0 | 0.0 |
| 2 | 0.5000000000 | 0.0 | 0.0 | 0.0 | 0.0 |
| 3 | 0.8333333333 | 0.7000000000 | 0.0 | 0.0 | 0.0 |
| 4 | 0.5833333333 | 0.6904761905 | 0.0 | 0.0 | 0.0 |
| 5 | 0.7833333333 | 0.6944444444 | 0.6932773109 | 0.0 | 0.0 |
| 6 | 0.6166666667 | 0.6924242424 | 0.6931057564 | 0.0 | 0.0 |
| 7 | C.7595238095 | 0.6935897436 | 0.6931633407 | 0.6931488693 | 0.0 |
| 8 | 0.6345238095 | 0.6928571429 | 0.6931399010 | 0.693146682C | 0.0 |
| 9 | 0.7456349206 | 0.6933473389 | 0.6931508286 | 0.6931473540 | 0.6931471961 |
| 10 | 0.6456349206 | 0.6930033417 | 0.6931451962 | 0.6931471119 | 0.6931471760 |

where the tenth partial sum, A(0,10), is correct to only one significant figure. However, iterate A(4,10) is correct to eight figures since Ln(2) = 0.6931471806.

In this thesis research, Shanks' accelerator has been tested extensively in conjunction with the simple parameterized contours of integration discussed in Chapters II and III.

Modification of the basic algorithm of Chapter III consisted of the elimination from the main program of the termination criterion for the numerical integration given by equation (16), and, instead, performing the integration for a finite number of terms. These addends of equation (16) were stored in a linear array and, subsequently, transformed, within the main program in accordance with equation (28).

The accelerator did not enhance the rate of convergence of the numerical inversion process for any of the cases investigated. In fact, if the number of terms of $A_n$, which are transformed, becomes sufficiently large so that the summation of the addend contributions of equation (16) approaches the correct result, the transformation will decrease the accuracy of the result as the denominator of equation (28) becomes small.

Appendix A contains the program listing for the simple parameterized contour of integration in conjunction with Shanks' accelerator.

Shanks [19] discusses transformations of higher order. These transformations may be worthy of investigation.

## VII. LOCATION OF FUNCTION SINGULARITIES

The criteria of Chapter II for the distortion of the Bromwich contour include the requirement that all of the singularities of a function f(s) be to the left of the contour. Heretofore, the discussion has assumed that the location of these singularities was known. Indeed, this is not, in fact, always the case. In this chapter we investigate a method of determining the location of the singularities of f(s) when such information is either not known or is difficult to obtain analytically.

To see how one can proceed without knowing where the singularities are, consider the simple case

$$f(s) = \frac{s}{s^2 + \alpha^2} \tag{30}$$

for the particular values of $\alpha = 0.125$ and $t = 2\pi$.

We use a simple parabolic contour given by equation (13) with B=1 and with A, which we denote by the symbol $x_o$ in the discussion to emphasize its actual significance, being given various values. If we plot the result of the inversion versus the starting position $x_o$, the result is as shown in Figure 3.

For values of $x_o$ less than zero, the result is 0.00. For $x_o$ in the range from about 0.2 to 2.0, the result is 0.707107. For $x_o$ larger than about 2.0, the result is near

0.7 but is sensitive to the precise value of $x_o$. For $x_o$ in the range from 0.0 to 0.2, the result exhibits a "spike" and a very significant change from 0.0 to 0.707107.

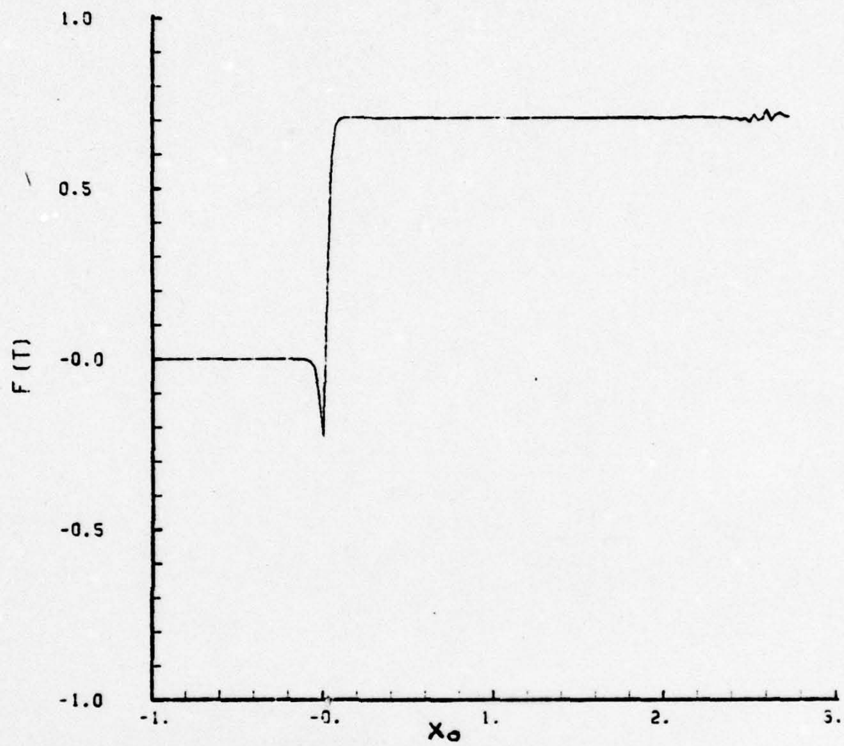



Figure 3. F(t) vs contour starting position for $\cos(\alpha t)$. Value of inverse of $\frac{s}{s^2+(.125)^2}$ for $t = 2\pi$ as a function of starting abscissa $x_o$.

We can explain this behavior as follows. For $x_o$ less than zero, the singularities are on the wrong side of the contour and we simply get the wrong answer. For $x_o$ in the neighborhood of 0.0 to 0.2, the contour passes so close to the singularity that the trapezoidal integration scheme is not accurate and experiences difficulty in getting an accurate value either for the right answer, 0.707107 or for the wrong answer 0.0. For $x_o$ in the range from 0.2 to 2.0 the singularities are on the correct side of the contour and we get the correct answer. For $x_o$ larger than 2.0, there are numerical difficulties, probably due to oscillation or to the effects of the exp(st) factor for s having large real part.

We have done the same thing with many other transforms and have found the same behavior. Our experience leads to this suggestion for cases where the user does not know the location of the singularities. Perform the calculation for a number of different values of $x_o$ and plot the results as in Figure 3. The correct result is the ordinate of the rightmost plateau, before oscillations or other numerical difficulties have had a chance to introduce inaccuracies. In the case illustrated in Figure 3, there were only two plateaus, one corresponding to all the singularities lying on the wrong side of the contour and one corresponding to all the singularities lying on the correct side of the contour. However, with more singularities, one might expect to have several plateaus, only the rightmost of which represents the correct result.

## VIII. THE HYPERBOLIC TANGENT

In Chapter II it was stated that great advantage accrues from distorting the Bromwich contour as long as the process of distortion does not cause the contour to cross over a singularity of f(s). In this chapter we investigate a typical case in which it is impossible to satisfy this condition.

The function

$$f(s) = \frac{1}{s} \tanh(\frac{\alpha s}{2}) \tag{31}$$

has an infinite sequence of simple poles on the imaginary axis at points $s = \pm(2n+1)\frac{\pi}{\alpha}$ . By other methods it may be shown that this function is the transform of the square wave function shown in Figure 4. In this chapter we investigate the extent to which our present methodology is capable of obtaining this square wave.

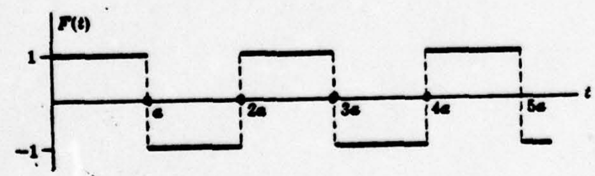



Figure 4. Square wave function.

The convergence of the numerical integration has been assured by bending the contour to the left in all other cases reported in this thesis. Accordingly, we do so in the present instance. This implies that we must pass between two poles as the contour bends to the left and all of the infinite number of poles above the point where the contour crosses the imaginary axis thus fall on the wrong side of the contour and are not included on the left. Thus, in effect, our procedure is finding the inverse Laplace transform of a substitute function which has only a finite number of poles coinciding in location with those of the given function f(s) and having the same residues. There is some reason to hope that the inverse of the substitute function will be essentially similar to that of the given function, at least for sufficiently small values of t.

Indeed we do find this to be the case. SUBROUTINE CURVE of the algorithm of Chapter II was modified to formulate a simple parameterized curve, which utilized the Bromwich contour for numerical integration in its initial segment, starting from a position $s=\gamma$ on the positive real axis and continuing until the elevation of a finite number of poles is reached. The second segment of the simple parameterized curve which was used consists of the parabola of equation (18). This parabolic arc allowed the integration contour to cross the imaginary axis between adjacent poles of the function f(s). Such a contour is indicated in Figure 5.

The termination criterion which was used in this case is identical to that described in Chapter III. All other program input values remained unaltered from the form of their

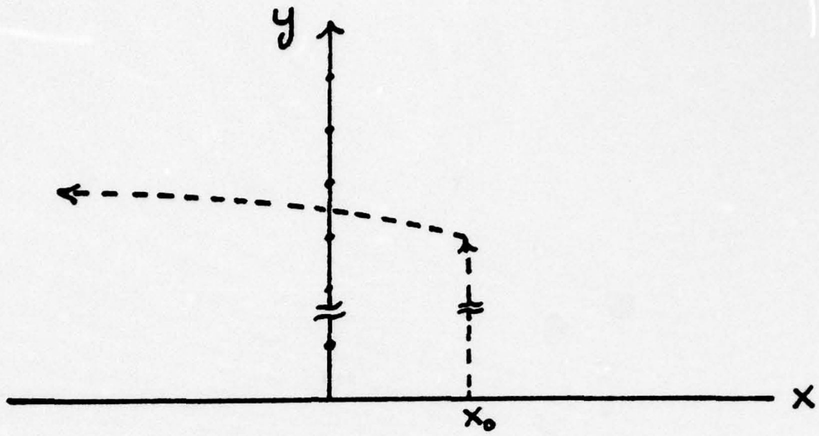



Figure 5. Contour of integration for the hyperbolic tangent. Special contour for inversion of $f(s) = \frac{1}{2} \tanh(\alpha s/2)$. It starts at $x_o$, rises vertically, and then follows a parabolic path between two poles.

previous presentation with the exception that the integer JJJJJ, which selected the number of function poles to be encompassed by the contour was added. The program listing appears in Appendix A.

The approximate solutions F(t) obtained by numerical inversion of the substitute function for the f(s) given by equation (31) are shown graphically below in Figures 6-9.

Figure 6 is a reproduction of five periods of the square wave of Figure 4 with $\alpha = 10.0$ and the first 100 poles of f(s) above the real axis to the left of the contour of integration. Figure 7 is a similar picture of the same F(t) with $\alpha = 5.0$ and the first 150 poles of f(s) above the real axis encompassed by the contour, with only one period of the wave form. Figures 8 and 9 are analogous to Figures 6 and 7,

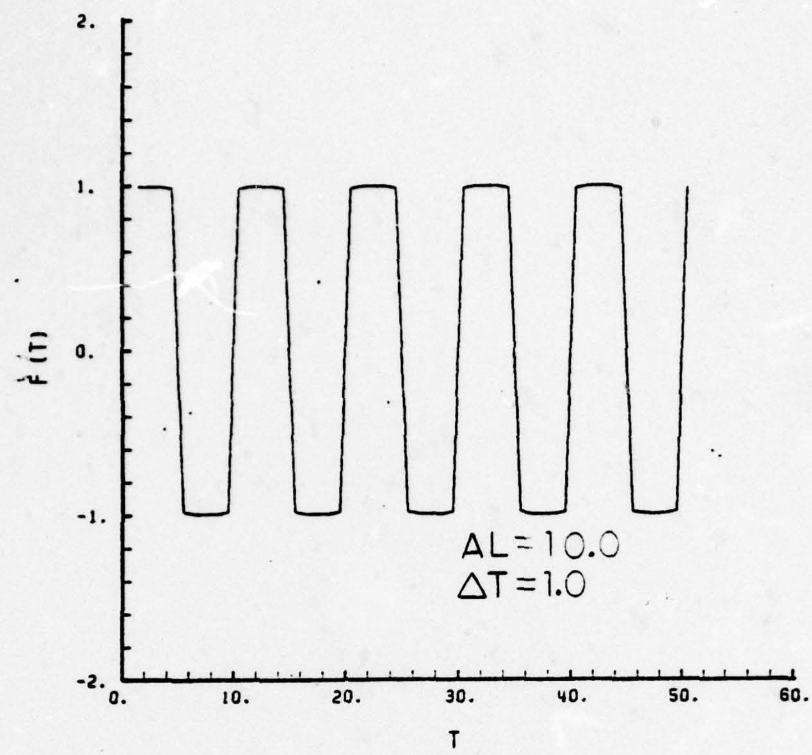



Figure 6. Numerical approximation of five cycles of the square wave function with 100 poles to the left of the contour.

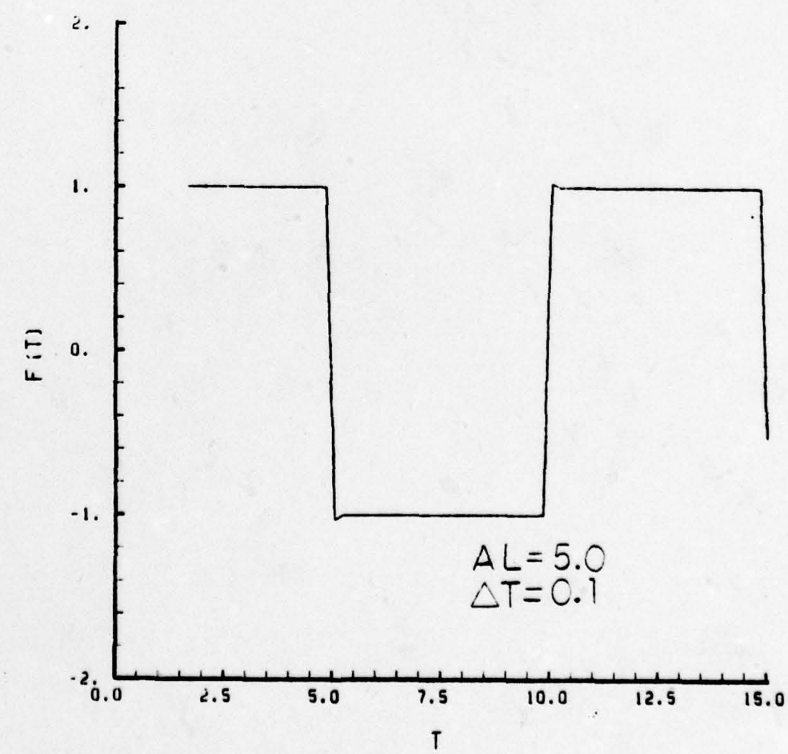



Figure 7. Numerical approximation of one cycle of the square wave function with 150 poles to the left of the contour.

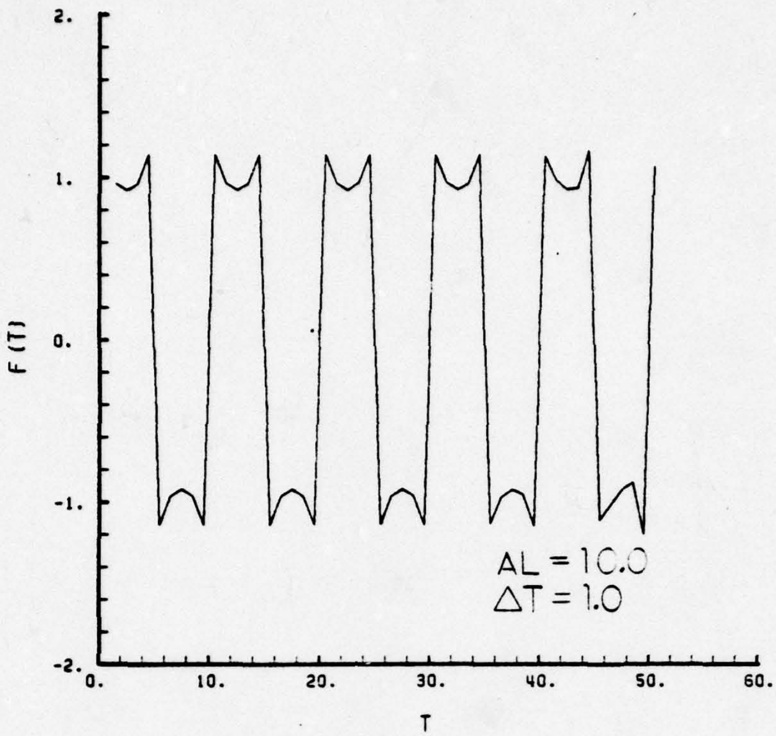



Figure 8. Numerical approximation of five cycles of the square wave function with 7 poles to the left of the contour.

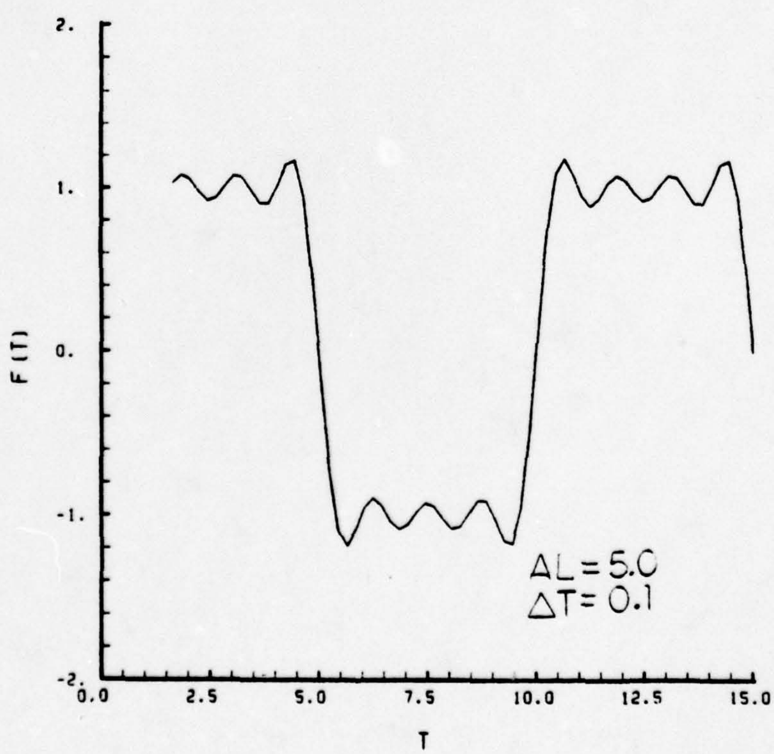



Figure 9. Numerical approximation of one cycle of the square wave function with 7 poles to the left of the contour.

respectively; however, only the first seven poles of f(s) above the real axis were included on the correct side of the contour of integration.

The reader should be aware that the increment of t used in the above series of graphs differed in each case. The inaccuracies shown in Figures 6-9 are partially due to the assumptions inherent within the employment of our algorithm, and, also, partially due to the fact that the plotter simply connects points sequentially. This can also be seen in the slope of the graphs at $t = \alpha$, $2\alpha$, etc., which is a function of the particular $\Delta t$ employed and not an inaccuracy of the algorithm.

Efforts to accomplish numerical inversion of the f(s) of equation (31) using the steepest descent contour were unsuccessful. The difficulty in using this contour appeared to be related to an adverse influence from the proximity of the function poles to the $v = 0$ contours near the imaginary axis. The algorithm was overtaxed in this vicinity and could not track the contour; consequently, further investigation was not attempted.

In summary, we conclude that the effect of substituting a function which has only a finite number of poles which are coincident in location with the first n poles of a function f(s), which possesses an infinite number of poles spaced incrementally along the imaginary axis, produces an inverse approximation to the inverse of the given function f(s). The accuracy obtained with such a substitution is increased as more of the function singularities are encompassed by the distorted contour.

# IX. FACTORS WHICH INFLUENCE ACCURACY

Heretofore, our discussion has included the treatment of contour integration, utilizing a simple parameterized curve for that purpose, from two perspectives. The first of these methods, discussed in Chapter III, incorporated a termination criterion which required a prescribed number, N, of successive integrand contributions each to have smaller magnitude than a specified epsilon. The second of these methods, discussed in Chapter VI, performed numerical integration for a finite number of points spaced along the contour with subsequent employment of Shanks' accelerator in an endeavor to enhance convergence.

The first of these methods, we have found, was highly successful when employed with the proper point spacing, and a very tight termination criterion, namely, a small epsilon and large N. The second method has not, in our investigations, been found to be successful, and need not be discussed further. The issue then becomes one of selecting the proper combination of step-size, epsilon, and N to provide an accurate and economical result.

In order to make such an appropriate selection of these parameters, it is necessary to examine the factors which influence accuracy. These factors include: the step-size along the contour, the termination criteria, and the oscillation.

Consider a simple case, which we have chosen, along a path for which the oscillation is insignificant. The accuracy in such a case is influenced, if the termination criteria are sufficiently stringent, only by the step-size along the contour of integration. Thus dealing with the transform

$$f(s) = \frac{s}{s^2 + \alpha^2} \qquad (32)$$

whose inverse is the cosine function, yields a plot of the logarithm of the percent error of F(t) versus the step-size, $\Delta$, as shown in Figure 10. In this case, the contour of integration was the simple parameterized curve of equation (18) with $A = 0.3$, $B = 1.0$, $\alpha = 0.125$ and $t = 2\pi$.

The sequences of points in this plot are all obtained by maintaining N constant (N=3) and plotting $\log_{10}$ percent error versus $\Delta$ with epsilon as a parameter.

Four such parameteric plots appear in the figure below corresponding to the values of epsilon (EPS) as shown in the legend.

Clearly the accuracy of the results may be seen to behave, in a general sense, in the manner shown in Figure 11, where $EPS_1 < EPS_2 < EPS_3 < EPS_4$.

When the termination criteria are tight, namely, when EPS is small and N is, at least, greater than one, the percent error is a narrowly banded, roughly linear function of the integration step size, $\Delta$, over a range from about $\Delta = 0.02$ to $\Delta = 0.2$. This corresponds to $EPS_1$ in Figure 11.

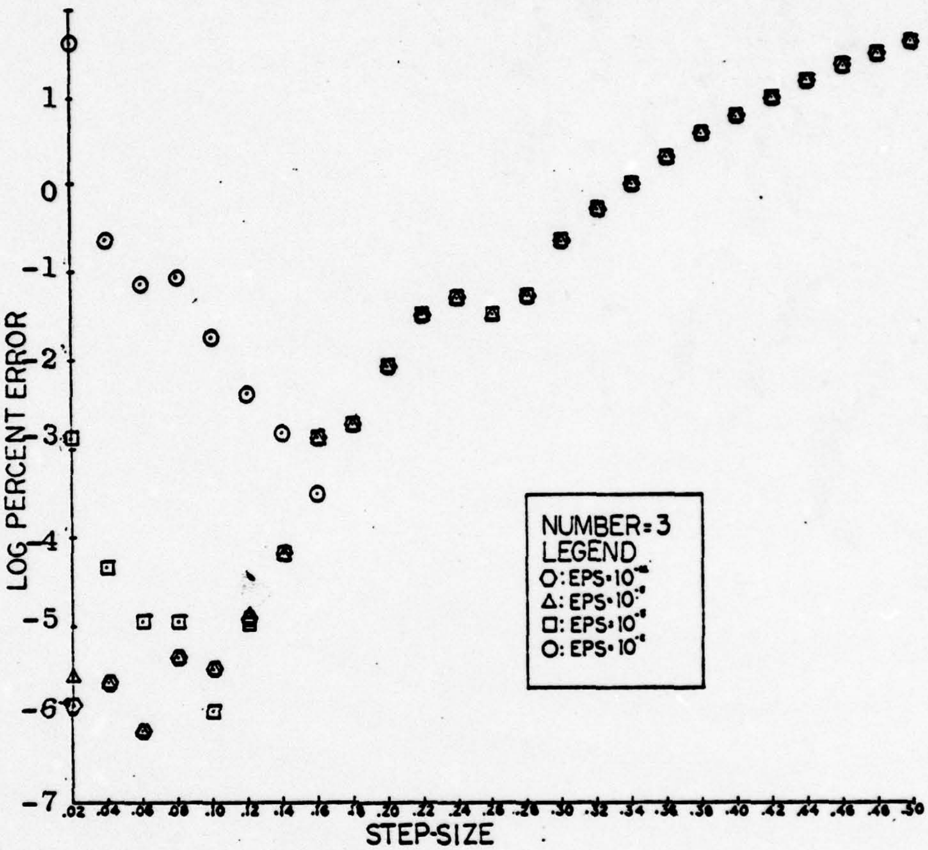



Figure 10. Logarithm of percent error versus step size for the cosine function.

If the termination criterion EPS is looser, as evidenced in other cases, shown schematically in Figure 11, the accuracy of the result cannot be improved indefinitely by decreasing, Δ. Rather, there is a point, which differs in each case, beyond which the accuracy is decreased as the step size is decreased. The behavior is complicated by the fact that the effects of changes in step-size and changes in termination criteria are not themselves disjoint. Making the step-size

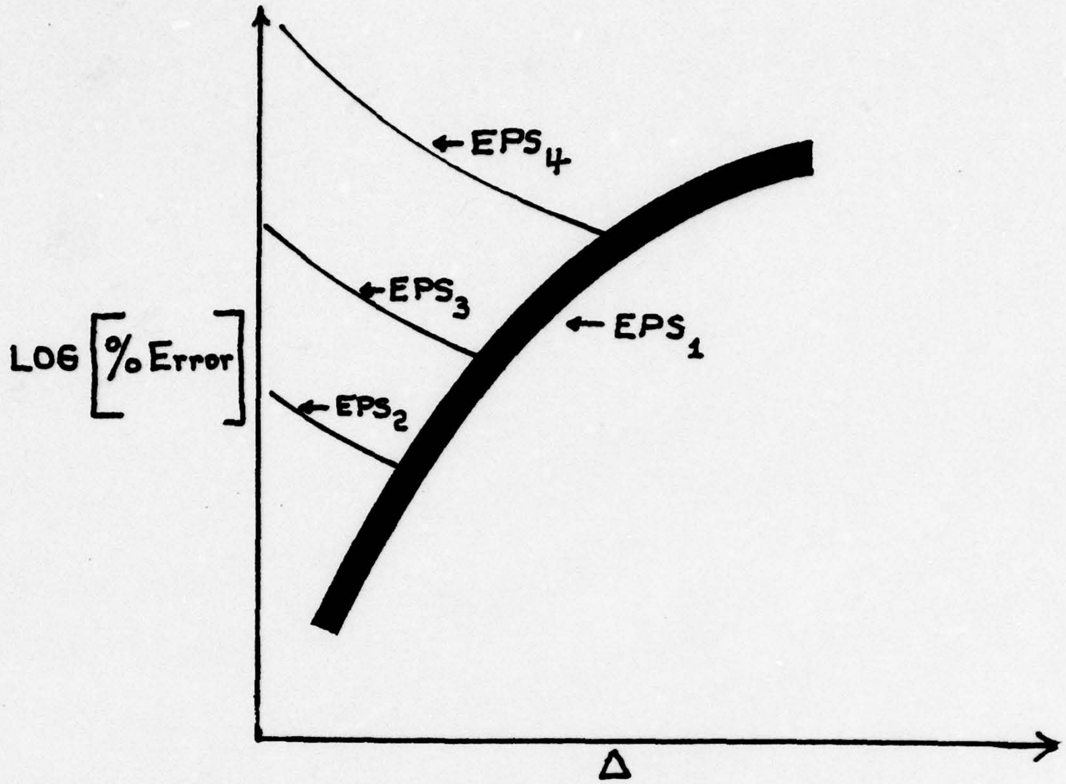



Figure 11. Schematic representation of accuracy factors. Behavior of the logarithm of percent error versus step size $\Delta$ as a function of the termination criterion EPS.

smaller results in meeting the termination criteria at an earlier point along the path of integration and thus may actually reduce overall accuracy.

One way of dealing with this matter, of course, is to revise the termination criteria so as to remove this interdependence. We simply have not experimented with using alternate forms for the termination algorithm. Instead, we suggest the following viewpoint to the user who wishes to assure that he obtains results with a specified accuracy.

He should first of all employ a very strict termination criterion and use a succession of rather small values of step size, and should also vary the starting point on the contour. In this way he should be able to establish an evaluation having even greater accuracy than he requires. Then, selecting what seems to be a good starting point and keeping to it, he should increase the step size until the result is no more accurate than he requires. He should observe how much saving in computer time (or in the number of calls to VALUE) this affords and should not increase step size beyond a reasonable point. Then he should loosen the termination criterion progressively, measuring the saving in computer time versus the possible loss of accuracy, stopping short of the point where he cannot rely on obtaining the required accuracy.

In many cases where numerical integration is employed to obtain a result whose correct value is Y, the numerically produced approximation y approximately satisfies a relation

$$y = Y + a\Delta^m \tag{33}$$

where a and m are unknown constants. The exponent m may frequently be established, once and for all, for the type of integration being used, and, for individual integrands, the correct value, Y, from two evaluations, $y_1$ obtained with $\Delta = \Delta_1$ and $y_2$ obtained with $\Delta = \Delta_2$, by using the extrapolation formula

$$Y = \frac{1}{\Delta_1 - \Delta_2} [\Delta_1 y_2 - \Delta_2 y_1] \tag{34}$$

We have investigated this method of obtaining improved accuracy for our inversions and have found that with the interdependence between termination criteria and step size, the numbers a and m were both rather large and difficult to establish - in other words, the extrapolation was not successful in producing improved values. Our conclusion was that it was more efficient from a computational point of view simply to use a sufficiently small step size and an appropriately matched set of termination criteria so as to be able to obtain an accurate answer without interpolation.

However, a suggestion for further study of this matter is made at the end of the next chapter.

## X. RESULTS, CONCLUSIONS, AND RECOMMENDATIONS

This thesis has demonstrated that numerical integration along a suitable contour in the complex plane, thus implementing the complex inversion integral formula, is an effective way of obtaining numerical values for the inverse Laplace transformation of a given function.

The success of the procedure, i.e., its efficiency, avoidance of inaccuracy due to oscillation, and its termination after including only a reasonable number of addends in the numerical sum, is related to the fact that the contours chosen bend to the left as they rise, thus taking advantage of the exp(st) factor in the integrand of the inversion formula.

Although there are some theoretical advantages to what we have called the "steepest descent" contour which follows a $v=0$ path, the algorithm which permits following such a path requires repeated evaluations of the integrand, many of which are not actually used in forming the summands for the numerical summation. We have found by investigating almost 100 cases for which known inverses are available in analytic form, that it is more efficient to use a simpler contour than the steepest descent path. By using what we have called a simple parameterized curve, we have devised a procedure which makes use of every evaluation of the integrand. We have found that the penalties of not following a steepest descent curve, namely an increase in oscillation of the integrand along the path, may be made quite tolerable by using any of a number of simple curves. In particular, most of our successful work has been with a simple parabola. Our algorithm, however, includes the output of information which can serve to alert the user to the possibility of inaccuracy due to oscillation and thus suggest to him that he might select a different form for the path.

We have even been successful in obtaining the inverse of a function which has an infinity of poles spaced along the imaginary axis. In doing so we violated the principle that all poles of the transform must be to the left of the contour. Nevertheless the results are quite satisfactory.

Our success and accuracy have been so gratifying that we venture to suggest that our method may prove to be an efficient alternate method for the evalution of exotic functions for which other methods are slowly convergent or involve series the coefficients in which are difficult to obtain. For example, case 92 in Appendix B shows the successful evaluation of the rather uncommon Struve function.

The original reason for attempting to employ numerical integration in the complex plane as a means of inverting a Laplace transformation was that alternate methods employed by Hiep and Zargary in conjugated heat transfer problems were simply not accurate enough. They led to physically impossible results with some temperatures in the media below sink temperature and others above source temperature.

this thesis we have not restudied these heat transfer problems and recommend that this be done using our methods of inversion. There is some reason to believe that the difficulty encountered by Hiep and Zargary will not be encountered if our method is used, but we are not prepared to substantiate this claim at this time.

Also it might be of use if the effects of varying the available parameters (shape of curve, spacing of points, starting point of curves, termination criteria, etc.) were to be investigated further so that a user would be better prepared to deal with indications of inaccurate inversion or inordinate requirements for computer time. Our own numerical experiments were invariably so happily successful that we have not encountered need for such information.

For functions f(s) for which it may be difficult to locate the singularities, we have shown that varying the starting point of a simple parameterized contour permits selecting an optimum contour in the sense that one can be assured that all singularities are to the left of the curve and also that the curve does not reach far enough to the right to impose numerical difficulties with large positive exponentials.

At the end of the preceding chapter we indicated that we did not find it profitable to employ extrapolation as a means of obtaining improved accuracy. However, this is probably worth looking at again, and our suggestions for doing so are as follows. First employ a termination criterion which is disjoint from the integration step size in the sense that the location, along the path of integration, of the point where termination occurs, is independent of step size. One way of doing this is to employ an epsilon in the termination criterion which is itself the result of multiplying a fixed, input epsilon by the step size. Then the extrapolation has a chance of being successful. So as to maintain optimum computational efficiency, one should use what is called Richardson extrapolation in which the evaluation points for the larger step size are themselves included among those for the smaller step size.

# APPENDIX A

## LISTING OF COMPUTER PROGRAMS AND SUBROUTINES

The computer programs and principal subroutines that we have prepared and tested during our investigation are all listed within this appendix. Section A-1 contains the user's instructions glossary, flowchart, and program listing for the simple parameterized curve. Section A-2 is a similar treatment for the steepest descent contour. Section A-3 contains the user's instructions and program listing for the simple parameterized curve procedure adapted for use with Shanks' accelerator. Section A-4 contains the same information for adaptation of the simple parameterized curve for use with the hyperbolic tangent function described in Chapter VIII.

It is hoped that sufficient detail has been provided to enable a user to adapt one of these programs to his purpose in an efficient manner. However, if additional insight is required, it may be necessary to refer to the chapter of this thesis in which the algorithm is developed. This is particularly true in the cases of Sections A-3 and A-4; the development in these sections has not been repeated where it is equivalent to that of Section A-1.

Functions DREAL and DIMAG and SUBROUTINES CPOWER and CLOG, if they are required, are listed within Chapter III, as is a typical example of SUBROUTINE VALUE. Additionally, all functions f(s) which have been investigated during this research have their applicable SUBROUTINE VALUE listed within Appendix B.

## SECTION A-1: THE SIMPLE PARAMETERIZED CURVE USER INSTRUCTIONS

This is the FORTRAN IV program for the numerical inversion of the Laplace transformation of a function f(s), which performs contour integration along a distorted Bromwich contour in the form of a simple parameterized curve. The following instructions are provided to assist the user in familiarizing himself with the program so that he can adapt it to his particular requirements.

1. SUBROUTINE CURVE is a user supplied group of instructions which formulates the numerical integration contour. The program listing presently contains a simple parabola, which may be utilized directly if desired.

2. SUBROUTINE VALUE is a user supplied subroutine which calculates the real and imaginary components for g(s) = exp(st)*f(s) = u(s) + iv(s). For examples of preparation of this subroutine the user is referred to the test cases of Appendix B.

3. The mandatory input variables to the main program are as follows:

   (a) AA - REAL*8 starting position (S=AA) on the real axis of the contour of integration. (Also denoted by A in equation (18) and $x_o$ in Chapter VII.)

   (b) AL - REAL*8 value of any constant associated with the function f(s). (Note: If the function f(s) involves more than one constant, the calling statements to SUBROUTINE VALUE must be modified accordingly.

(c) DELP-REAL*8 increment of the p-parameter leading to spacing of points along the contour of integration.

(d) T - REAL*8 time for which F(t) is desired.

4. Input parameters which may be altered by the user, at his discretion, are as follows:

(a) EPS - REAL*8 tolerance parameter for the termination of numerical integration. The program listing presently contains a value of 1.D-11.

(b) NUMBER - INTEGER*4 number of successive addend contributions less than the specified value of EPS for which numerical integration is terminated.

5. The output variables which are printed as output by program are defined below:

(a) AA starting position on the real axis

(b) SUM the value of F(t) obtained by numerical inversion of f(s). (cf. equation (16))

(c) SUMA the absolute value of the summation of the the addend contributions of the numerical integration. (cf. equation (14))

(d) X the final value of x on the contour at which integration was terminated

(e) Y the final value of y on the contour at which integration was terminated

(f) LOSC the number of sign changes between successive addends encountered during numerical integration

(g) MMMMM the total number of calls made to SUBROUTINE VALUE during the numerical integration process.

6. The output of these variables is as follows:

AA, SUM, SUMA, X, Y, LOSC, MMMMM

7. The time (seconds) required to perform the numerical integration is printed on a line immediately preceding the items listed above.

8. As previously described, the program will perform numerical integration along one simple parameterized curve. If the user desires, the program may be modified to "sweep" the contour of integration over a range of starting values, AA, (cf. Chapter VII). This may be quickly accomplished in the following manner.

(a) Replace the value of the integer KSTART from 1 to the number of contour integrations desired.

(b) Replace the REAL*8 value of AA with the following statement

AA = XSTART+AK*XINCRM

where

XSTART = the smallest starting value of the contour of integration the user desires

XINCRM = the $\Delta x$ between starting positions of successive contours of integration

must also be supplied.

Thus, the program will perform numerical inversion along KSTART contours of integration which differ only in their starting positions along the x axis. The user may then observe increases or jumps in the value of the output variable SUM between successive integrations, enabling him to ascertain the locations of function poles. The utility of this procedure is discussed in depth in Chapter VII.

9. If SUBROUTINES CPOWER and/or CLOG are called by SUBROUTINE VALUE in conjunction with use by this program, NCALL and/or MCALL must be dimensioned and initialized to zero within the main program and added to the calling statements to SUBROUTINE VALUE. Additionally, any powers, other than integer values, must be entered within the main program and passed within the calling statements to SUBROUTINE VALUE.

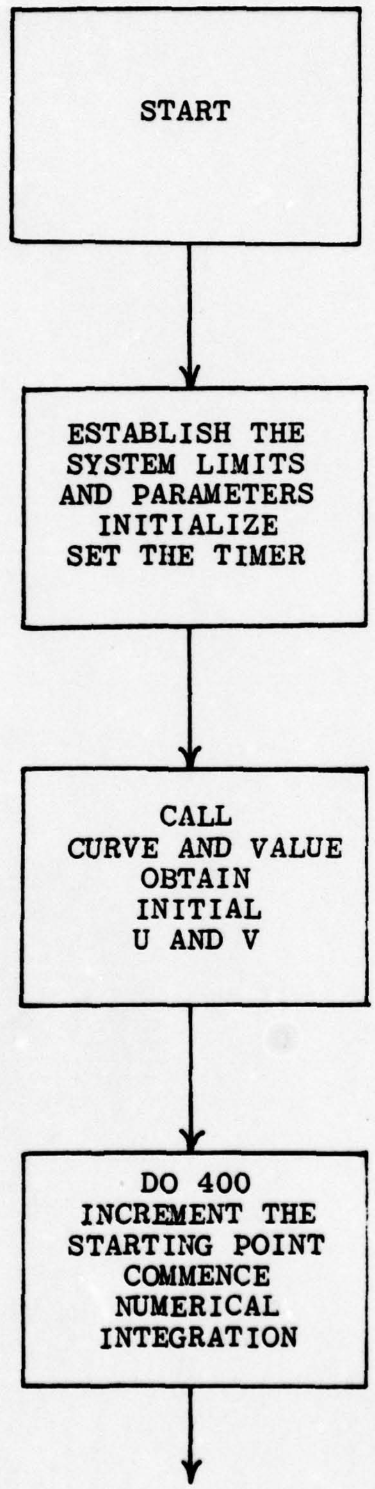
## GLOSSARY OF VARIABLE NAMES

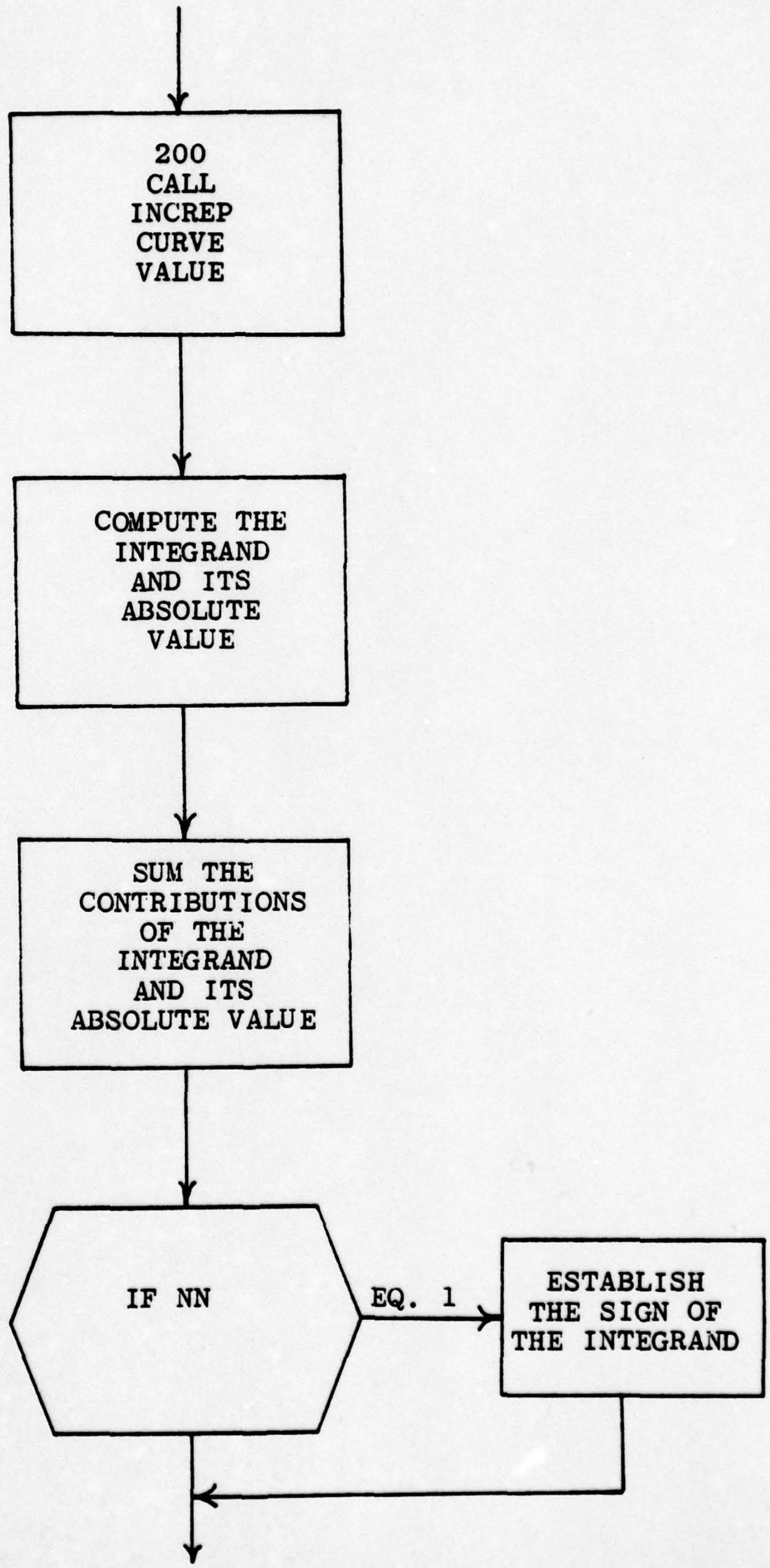
| VARIABLE | DESCRIPTION | LEGEND |
|---|---|---|
| AA | starting position on the real axis of the simple parameterized curve | 2 |
| ADD | computed value of the addend of the numerical integration | 1 |
| ADDA | absolute value of the addend of the numerical integration | 1 |
| ADDDLD | saved value of the previous addend contribution | 1 |
| AL | constant associated with f(s) | 2 |
| DELP | increment of point spacing along the simple parameterized curve | 2 |
| EPS | tolerance parameter for termination of the numerical integration process | 4 |
| KOUNT | integer counter utilized to terminate the numerical integration process | 1 |
| KSTART | integer counter for the number of times the numerical integration is desired to be performed, using different starting positions | 4 |
| LOSC | integer counter utilized to record the number of sign changes between successive addends | 3 |
| MMMMM | integer counter to record the total number of calls made to SUBROUTINE VALUE during the numerical integration | 3 |
| NN | integer counter to record the total number of addend contributions | 1 |
| NUMBER | integer input of the desired number of successive addend contributions less than a specified epsilon for termination | 4 |
| P | parameter of the simple parameterized curve | 1 |
| PROD | flag utilized to detect oscillation of the integrand | 1 |

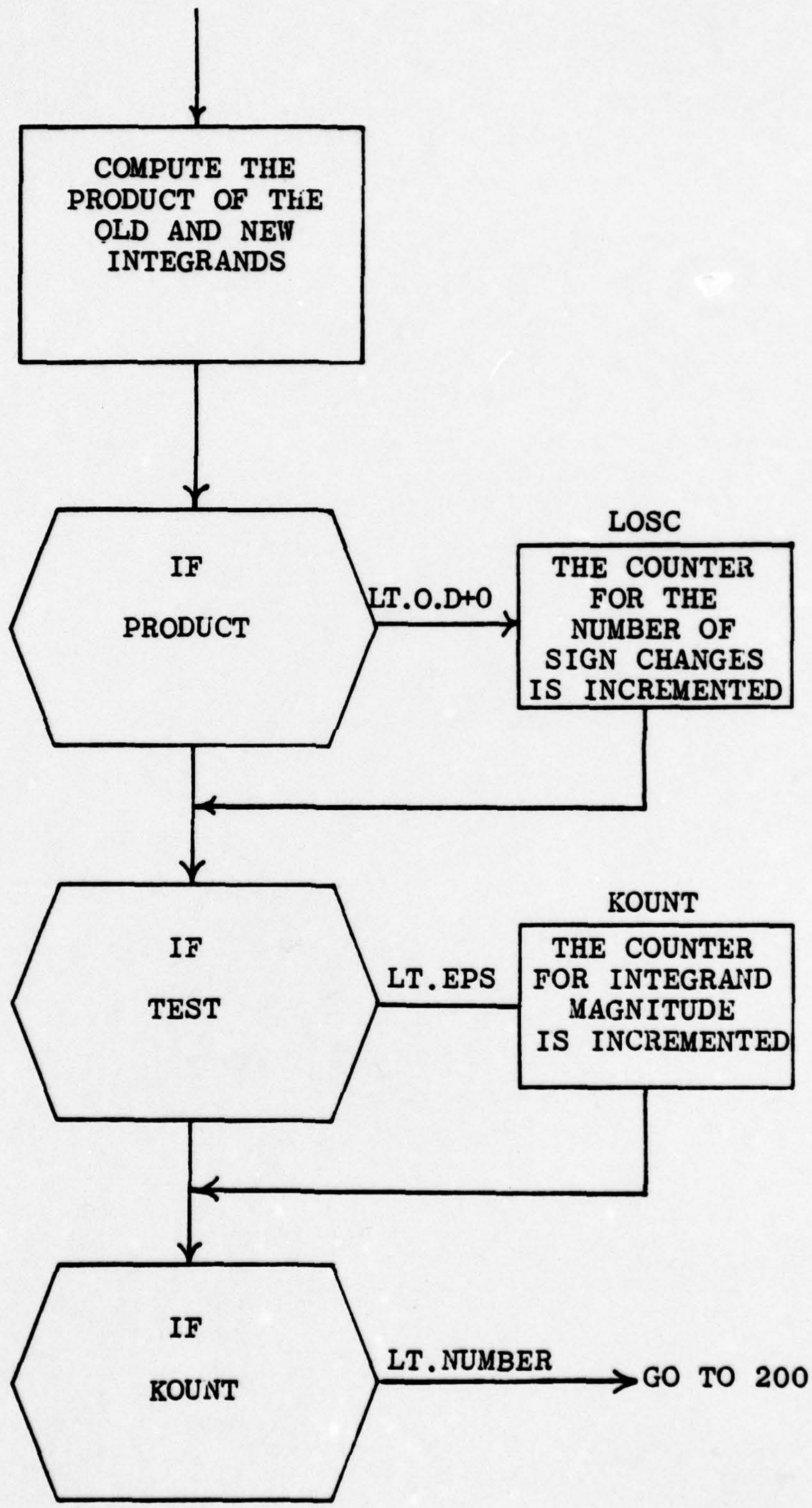
| | | |
|---|---|---|
| SUM | total addend contribution for the numerical integration process | 3 |
| SUMA | absolute value of the total addend contribution for the numerical integration process | 3 |
| T | time for which F(t) is to be evaluated | 2 |
| TEST | flag for termination of the numerical integration process | 1 |
| U | real part of the function g(s) | 1 |
| UO | previous value of U | 1 |
| V | imaginary part of the function g(s) | 1 |
| VO | previous value of V | 1 |
| X | real part of s | 1 |
| XO | previous value of X | 1 |
| Y | imaginary part of s | 1 |
| YO | previous value of Y | 1 |

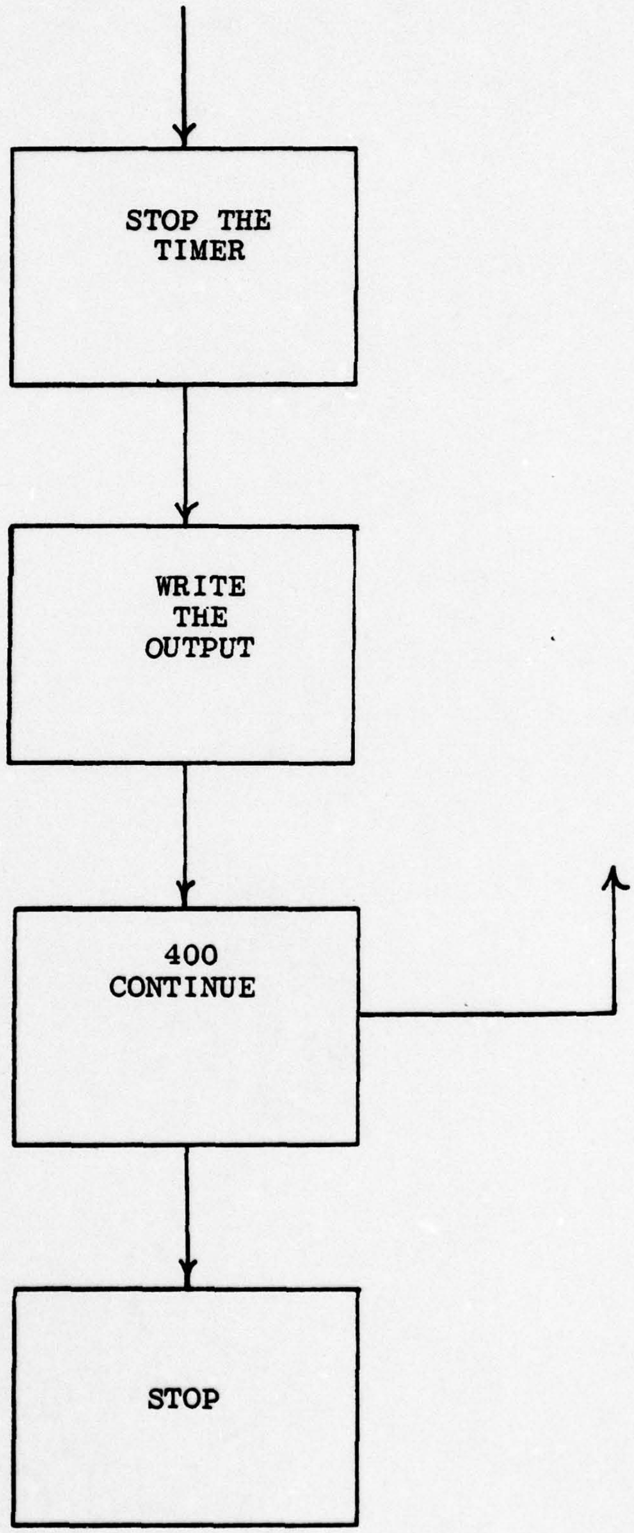
LEGEND

1. No action required by the user
2. Mandatory user input
3. Appears as program output
4. May be altered by the user at his discretion


START
ESTABLISH THE SYSTEM LIMITS AND PARAMETERS INITIALIZE SET THE TIMER
CALL CURVE AND VALUE OBTAIN INITIAL U AND V
DO 400 INCREMENT THE STARTING POINT COMMENCE NUMERICAL INTEGRATION


200
CALL
INCREP
CURVE
VALUE
COMPUTE THE
INTEGRAND
AND ITS
ABSOLUTE
VALUE
SUM THE
CONTRIBUTIONS
OF THE
INTEGRAND
AND ITS
ABSOLUTE VALUE
IF NN
EQ. 1
ESTABLISH
THE SIGN OF
THE INTEGRAND


COMPUTE THE PRODUCT OF THE OLD AND NEW INTEGRANDS
IF PRODUCT
LT.O.D+0
LOSC
THE COUNTER FOR THE NUMBER OF SIGN CHANGES IS INCREMENTED
IF TEST
LT.EPS
KOUNT
THE COUNTER FOR INTEGRAND MAGNITUDE IS INCREMENTED
IF KOUNT
LT.NUMBER
GO TO 200


STOP THE
TIMER
WRITE
THE
OUTPUT
400
CONTINUE
STOP

```
C     THIS IS THE PROGRAM TO PERFORM NUMERICAL INVERSION OF THE LAPLACE
C     TRANSFORMATION OF A FUNCTION USING THE SIMPLE PARAMETERIZED CURVE
      IMPLICIT REAL*8 (A-H,O-Z)
      CALL ERRSET (207,256,-1,1,1,209)                          LEGEND
      NUMBER = 5                                                  4
      KSTART = 1                                                  4
      TWOPI = 4.D+0*DARSIN(1.D+0)
      AL = 1.25D-1                                                2
C
C     #################################################################
C     #                                                               #
C     #  THE OUTER LOOP MAY BE UTILIZED TO SWEEP THE CONTOUR STARTING #
C     #POSITION OVER A DESIRED RANGE AS DESCRIBED IN THE USER INST.   #
C     #                                                               #
C
      DO 400 KKK=1,KSTART
      ADD = 0.D+0
      ADDA = 0.D+0
      SUM = 0.D+0
      SUMA = 0.D+0
      LOSC = 0
      MMMMM = 0
      AK = KKK
      AA = 3.D-1                                                  2
      T = TWOPI                                                   2
      P = 0.D+0
      NN = 0
      KOUNT = 0
      EPS = 1.D-11                                                4
      DELP = 1.D-1                                                2
      CALL CURVE (AA,P,XO,YO)
      CALL VALUE (XO,YO,T,AL,UO,VO,MMMMM)
      TIME = 0.D+0
      CALL SETIME
C
C     $$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
C     $                                                               $
C     $  THE FOLLOWING LOOP PERFORMS THE NUMERICAL INTEGRATION        $
  200 NN = NN+1
      CALL INCREP (P,DELP)
      CALL CURVE (AA,P,X,Y)
      CALL VALUE (X,Y,T,AL,U,V,MMMMM)
      ADD = ((V+VO)*(X-XO)+(U+UO)*(Y-YO))/TWOPI
      IF (NN.EQ.1) ADDOLD=ADD
      PROD = ADD*ADDOLD
      IF (PROD.LT.0.D+0) LOSC=LOSC+1
      ADDA = DABS(ADD)
      SUM = SUM+ADD
      SUMA = SUMA+ADDA
C
C     *****************************************************************
C     *                                                               *
C     *  THIS IS THE TERMINATION CRITERION SEGMENT OF THE ALGORITHM   *
C     *                                                               *
      TEST = DABS(ADD)
      IF (TEST.LT.EPS) KOUNT=KOUNT+1
      IF (TEST.GE.EPS) KOUNT=0
      CALL GETIME (IET)
      EL = DFLOAT(IET)*2.6D-5
      TIME = TIME+EL
      CALL SETIME
      IF (KOUNT.EQ.NUMBER) GO TO 300
C     *                                                               *
C     *****************************************************************
C
      XO = X
      YO = Y
      UO = U
      VO = V
      ADDOLD = ADD
      GO TO 200
C     $                                                               $
C     $$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
C
```

```
  300 CONTINUE
      WRITE (6,500) NN,TIME
      WRITE (6,600) AA,SUM,SUMA,X,Y,LOSC,MMMMM
  400 CONTINUE
C
C     *                                                                 *
C     *******************************************************************
C
      STOP
C
C
  500 FORMAT (1X,'TOTAL ELAPSED TIME AFTER ',I5,' ITERATIONS IS ',E20.5)
  600 FORMAT (1X,1P5E20.5,2I8,/)
      END
      SUBROUTINE CURVE (A,P,X,Y)
      IMPLICIT REAL*8 (A-H,O-Z)
      X = A-(P*P)
      Y = P
      RETURN
      END
      SUBROUTINE INCREP (P,DELP)
      IMPLICIT REAL*8 (A-H,O-Z)
      P = P+DELP
      RETURN
      END
```

## SECTION A-2: THE STEEPEST DESCENT CONTOUR USER INSTRUCTIONS

This is the FORTRAN IV program for the numerical inversion of the Laplace transformation of a function f(s), which performs contour integration along the steepest descent path in the complex plane. The following instructions are provided to assist the user in familiarizing himself with the program so that he can adapt it to his particular requirements.

1. SUBROUTINE XMARCH is a subroutine which tracks a specified v=0 contour (steepest descent) in the complex plane, once this contour has been intersected for the first time. As shown in the program listing, the following two equations

```
X = XHOLD - 1.D-1*DCOS(THETA)
Y = YHOLD + 1.D-1*DSIN(THETA)
```

march the numerical integrations in the favorable direction along the contour. The equations are parametric in THETA, which is the search angle provided by SUBROUTINE ROTATE. The 1.D-1 is a polar radius which may be altered to a larger or smaller increment by the user, or accepted as it appears.

2. SUBROUTINE ROTATE increments the angular search angle THETA in order to locate a new point on the v=0 contour. The angle is presently incremented by 5.D-2 radians with each call to the subroutine. This may be altered as required.

3. SUBROUTINE VALUE is a user supplied subroutine which calculates the real and imaginary components for $g(s) = \exp(st) \times f(s) = u(s) + iv(s)$. For examples of preparation of this

subroutine, the user is referred to Appendix B.

4. The mandatory input variables to the main program are listed as follows:

(a) AA - REAL*8 starting position (s=AA) on the real axis

(b) AL - REAL*8 value of any constant associated with the function f(s). (Note: If the function f(s) involves more than one constant, the calling statements to SUBROUTINE VALUE must be modified accordingly.)

(c) DELY - REAL*8 increment of the point spacing along the Bromwich contour. (Note: to alter the point spacing along the v=0 contour, the user is referred to 1 above.)

(d) NREG - INTEGER*4 input value which specifies the first, second, third, etc. v=0 contour above the positive real axis, which is to be followed in the numerical integration process.

(e) T - REAL*8 time for which F(t) is to be evaluated.

5. Input parameters which may be altered by the user, at his discretion, are listed below as follows:

(a) EPS - REAL*8 tolerance parameter for the termination of numerical integration. The program listing presently contains a value of 1.D-11.

(b) NUMBER - INTEGER*4 number of successive addend contributions less than the specified value of EPS for which numerical integration is terminated.

6. The output variables which are received from the program are defined below:

(a) MMMMM - the number of calls made to SUBROUTINE VALUE

(b) SUM - the total addend contribution for the numerical integration

(c) SUMA - the absolute value of the total integrand contribution for the numerical integration

(d) EL - the computation time in seconds required for the process

7. A sample program output is as follows

```
694                              MMMMM     } at end of vertical
0.724724D+0   0.324972D+1        SUM,SUMA  } section
0.707107D+0   0.326734D+1        SUM,SUMA  } at end of
4027                             MMMMM     } computation
0.71090D+1                       EL        }
```

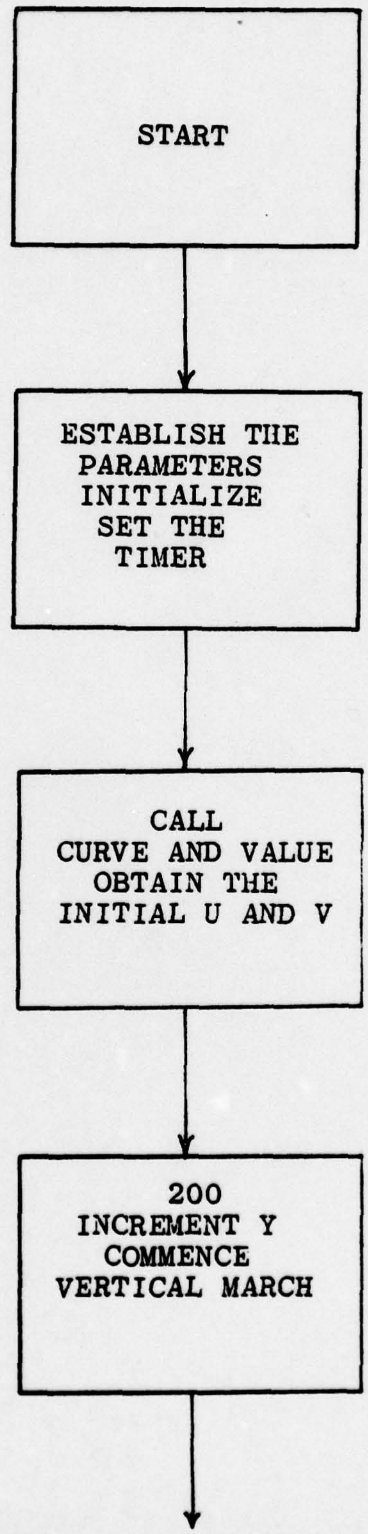
## GLOSSARY OF VARIABLE NAMES

| VARIABLE | DESCRIPTION | LEGEND |
|---|---|---|
| AA | starting position on the real axis of the Bromwich contour segment of the integration | 2 |
| ADD | computed value of the addend of numerical integration | 1 |
| ADDA | absolute value of the addend of numerical integration | 1 |
| AL | constant associated with f(s) | 2 |
| CHECK | flag to ascertain when a v=0 contour crossing has occurred | 1 |
| DELY | increment of point spacing along the Bromwich contour segment of numerical integration | 2 |
| EPS | tolerance parameter for termination of numerical integration | 4 |
| EXAM | flag to ascertain when a contour crossing has occurred | 1 |
| J | integer counter used internally for testing contour regions | 1 |
| KOUNT | integer counter utilized to terminate the numerical integration process | 1 |
| KSIGN | integer counter utilized to record the number of v=0 contour levels crossed | 1 |
| LEVEL | integer flag assigned to regions of positive and negative v | 1 |
| MMMMM | integer counter for the number of calls made to SUBROUTINE VALUE during each segment of the numerical integration process | 3 |
| NREG | the desired v=0 contour level above the positive real axis which is to be tracked | 2 |
| NUMBER | integer input for the number of successive addend contributions less than the tolerance parameter required for termination | 4 |

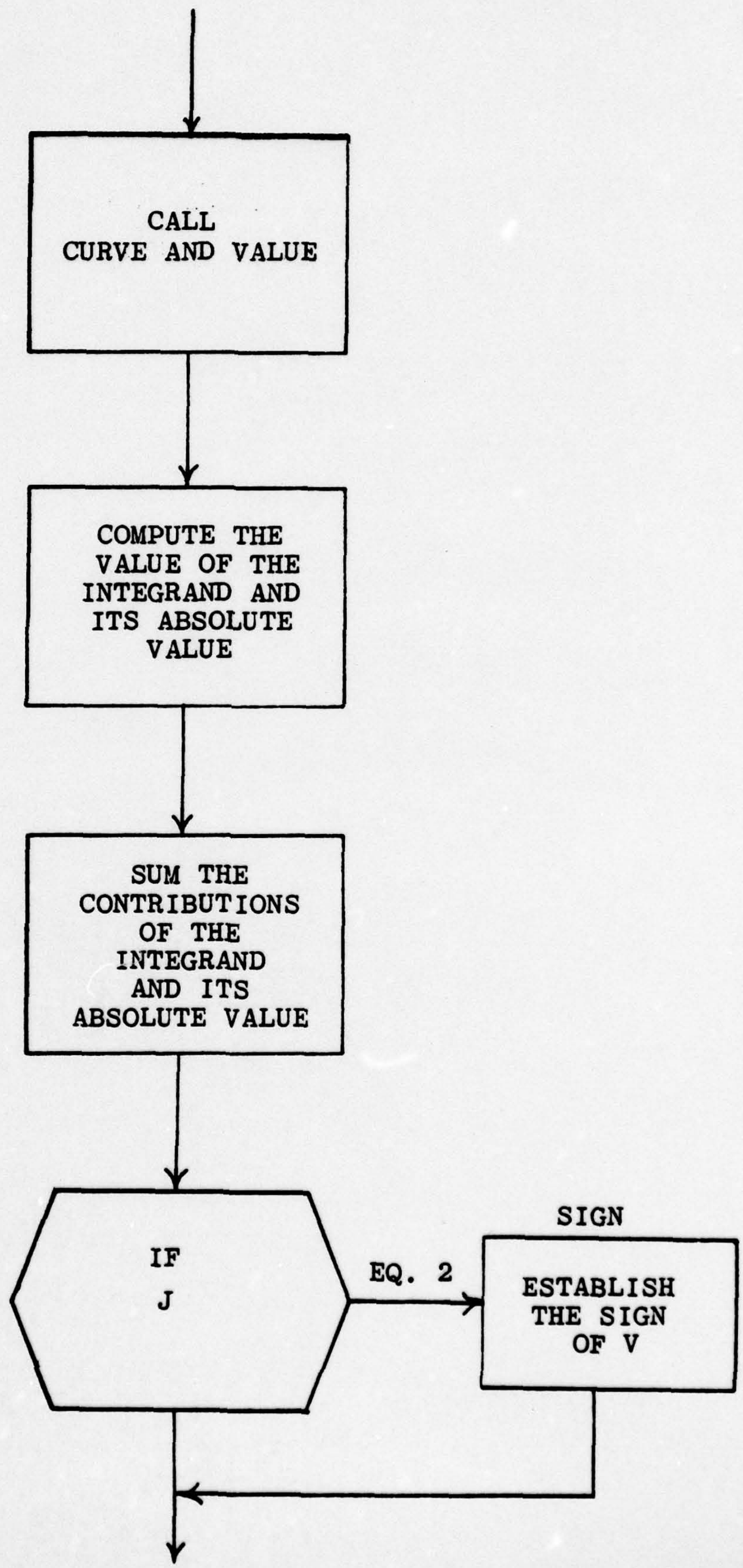
| | | |
|---|---|---|
| SUM | total addend contribution for the numerical integration process | 3 |
| SUMA | absolute value of the total addend contribution over the contour of integration | 3 |
| T | time for which F(t) is to be evaluated | 2 |
| TERM | termination parameter for the absolute value of the addend | 1 |
| TEST | Flag to ascertain when a contour crossing has occurred | 1 |
| THETA | search angle used in the contour tracking algorithm | 1 |
| THETAO | previous value of THETA | 1 |
| THNEW | predicted angular location of a new point on the contour from a known previous point. | 1 |
| THONE | coordinate utilized in linear interpolation | 1 |
| THTWO | coordinate utilized in linear interpolation | 1 |
| U | real part of g(s) | 1 |
| UHOLD | previous value of U | 1 |
| V | imaginary part of g(s) | 1 |
| VHOLD | previous value of V | 1 |
| VNEW | actual value of the imaginary part of g(s) for the predicted v=0 location | 1 |
| VOLD | previous value of V | 1 |
| X | real part of s | 1 |
| XHOLD | previous value of X | 1 |
| Y | imaginary part of s | 1 |
| YNEW | predicted elevation of the desired v=0 contour level | 1 |

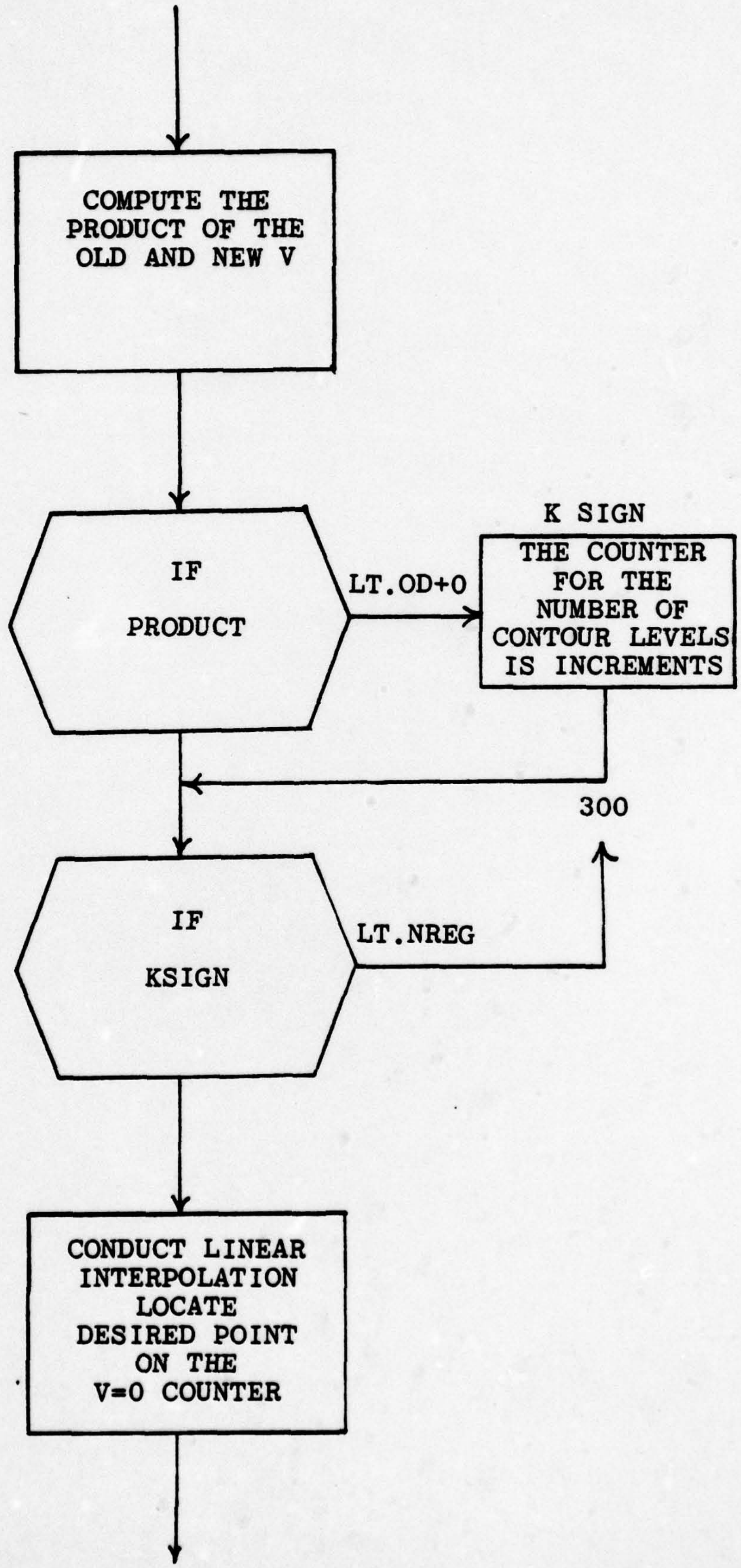
## LEGEND

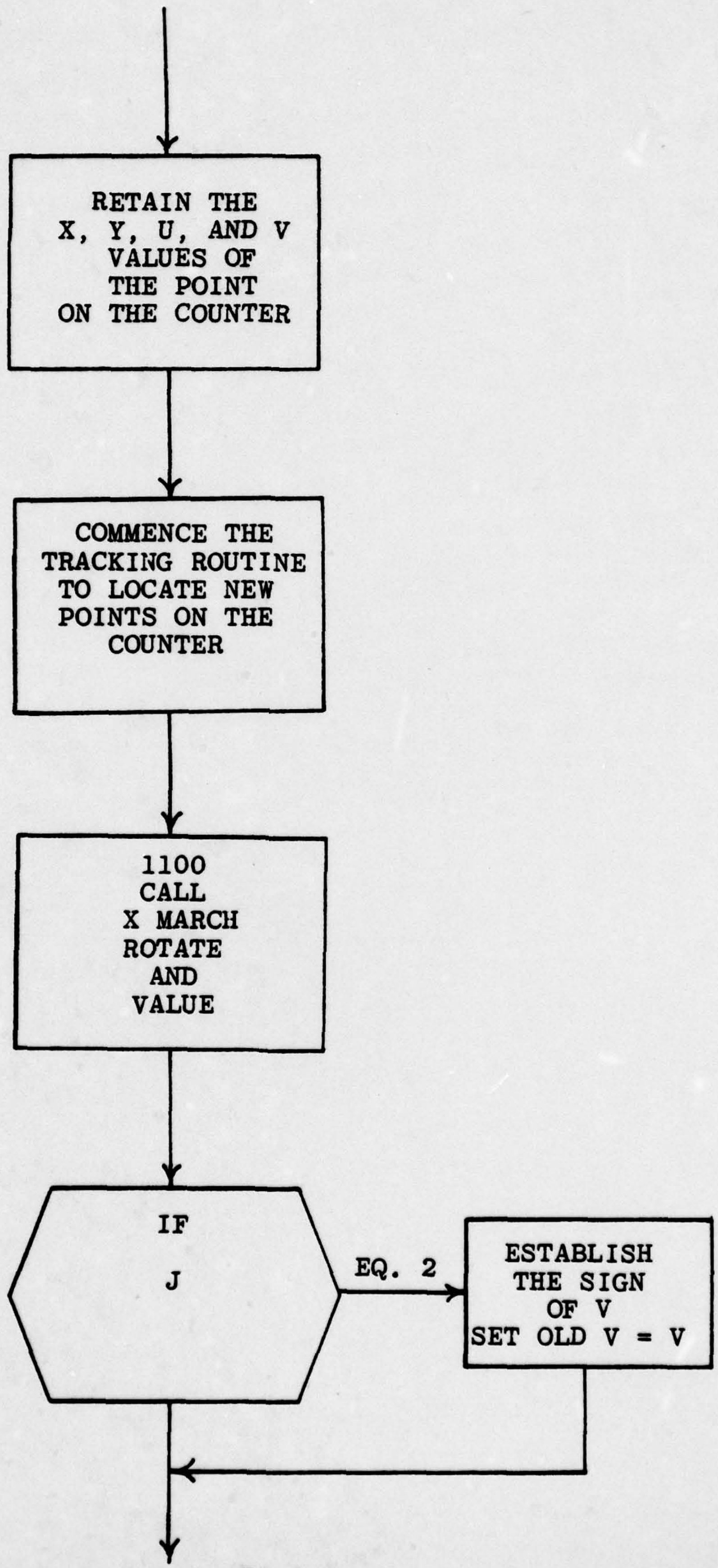
1. No action required by the user
2. Mandatory user input
3. Appears as program output
4. May be altered by the user at his discretion

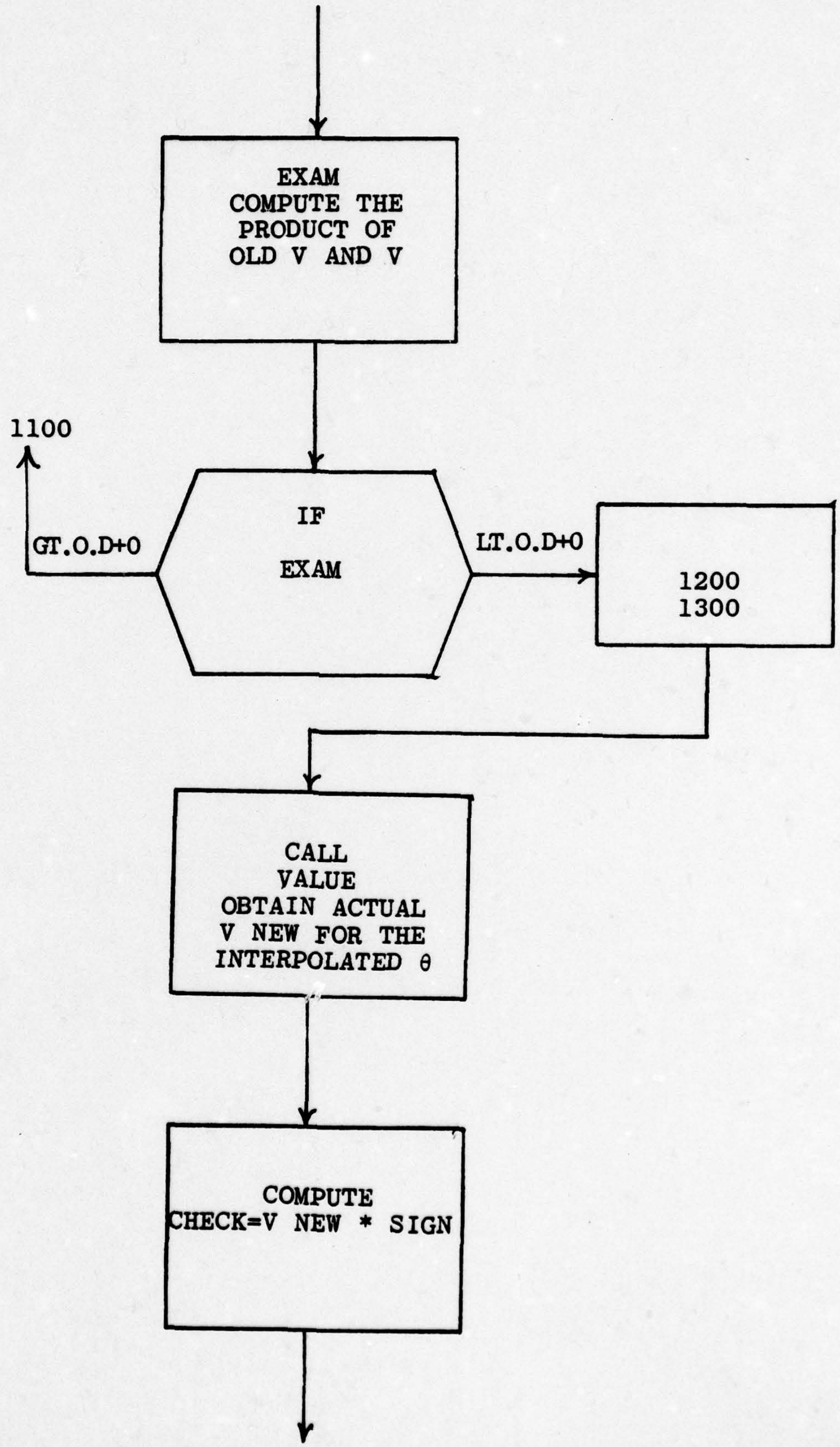


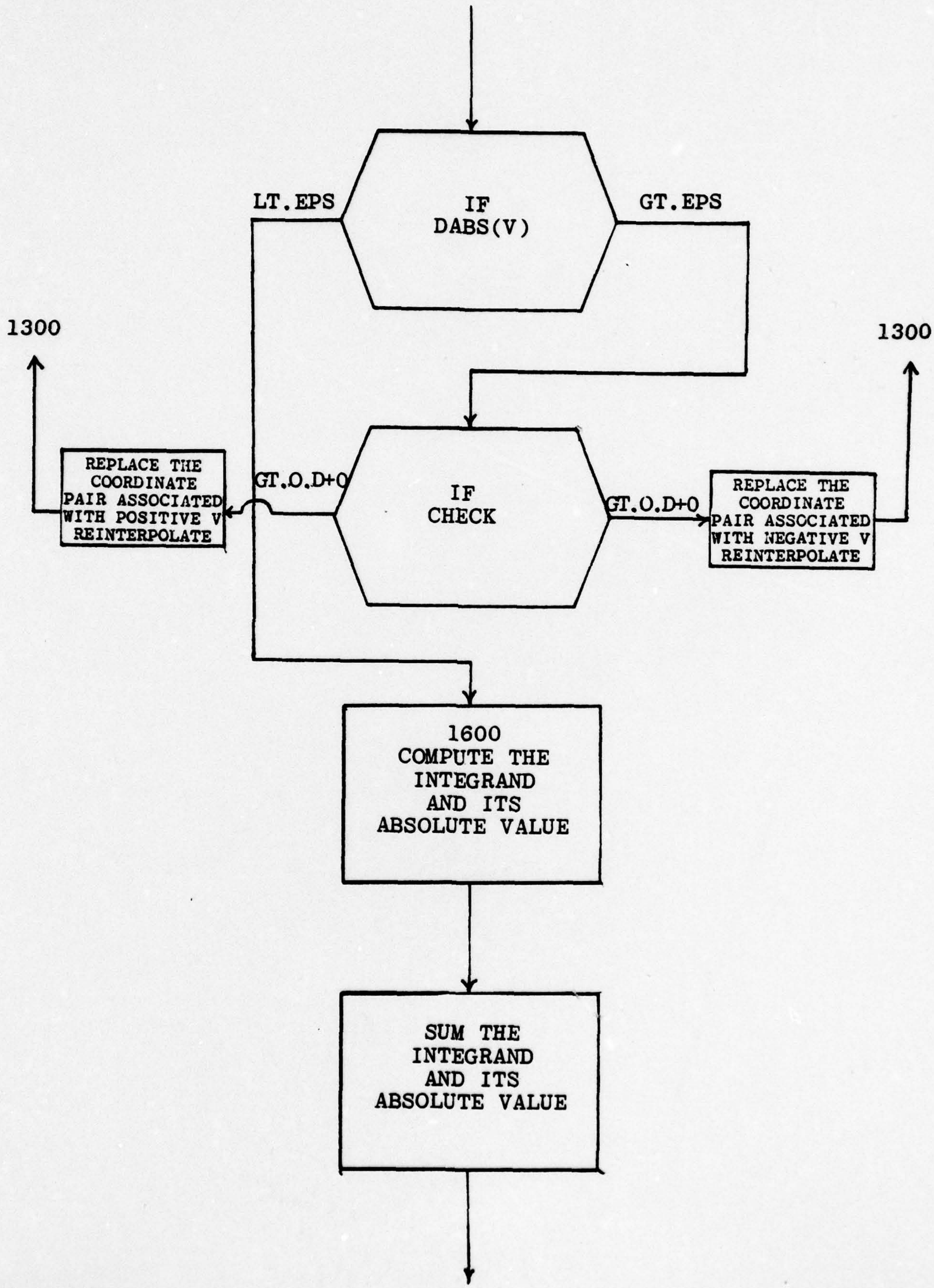
START
ESTABLISH THE PARAMETERS INITIALIZE SET THE TIMER
CALL CURVE AND VALUE OBTAIN THE INITIAL U AND V
200 INCREMENT Y COMMENCE VERTICAL MARCH

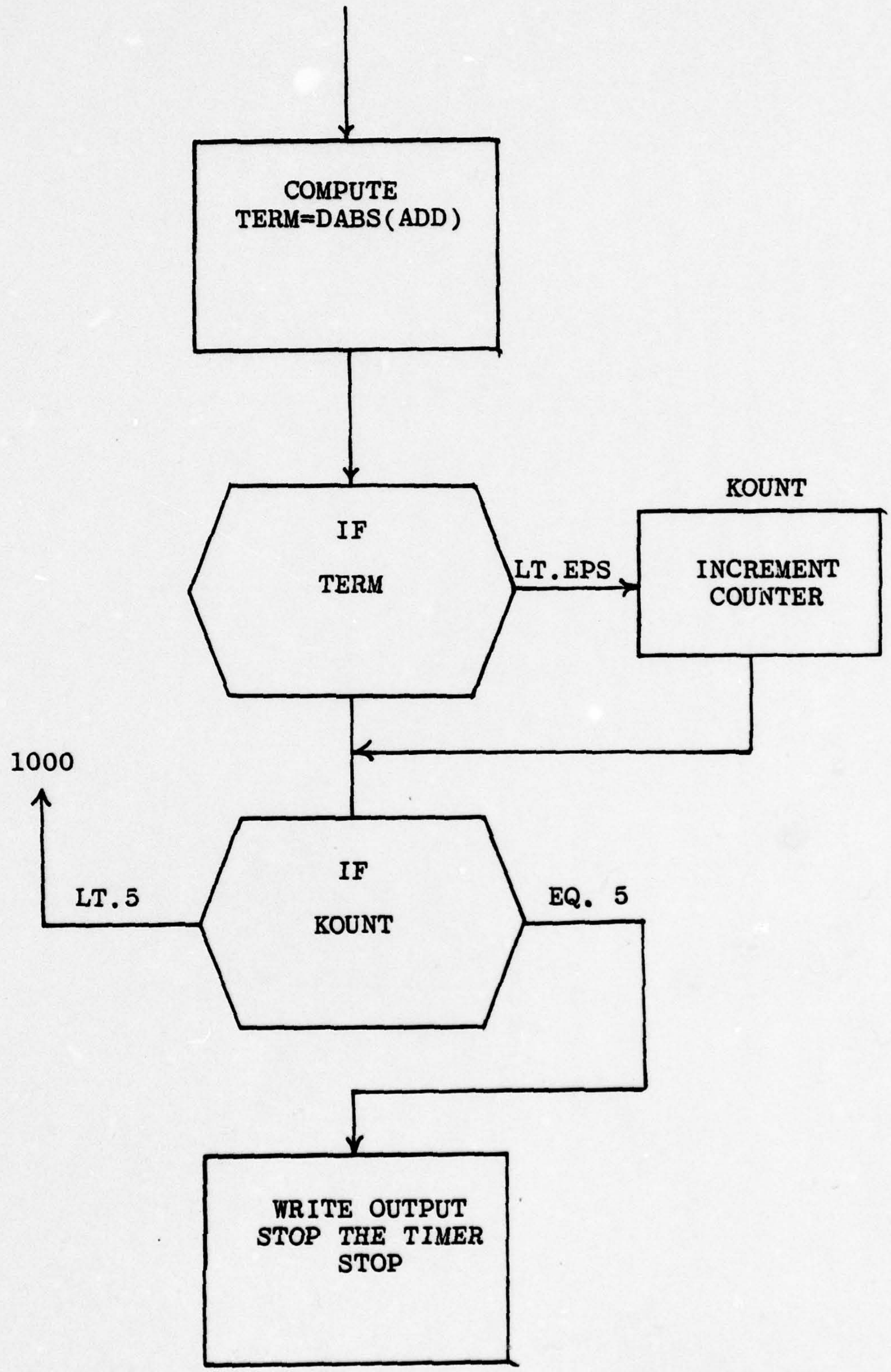

CALL
CURVE AND VALUE
COMPUTE THE
VALUE OF THE
INTEGRAND AND
ITS ABSOLUTE
VALUE
SUM THE
CONTRIBUTIONS
OF THE
INTEGRAND
AND ITS
ABSOLUTE VALUE
IF
J
EQ. 2
SIGN
ESTABLISH
THE SIGN
OF V


COMPUTE THE PRODUCT OF THE OLD AND NEW V
IF
PRODUCT
LT.OD+0
K SIGN
THE COUNTER FOR THE NUMBER OF CONTOUR LEVELS IS INCREMENTS
300
IF
KSIGN
LT.NREG
CONDUCT LINEAR INTERPOLATION LOCATE DESIRED POINT ON THE V=0 COUNTER


RETAIN THE
X, Y, U, AND V
VALUES OF
THE POINT
ON THE COUNTER
COMMENCE THE
TRACKING ROUTINE
TO LOCATE NEW
POINTS ON THE
COUNTER
1100
CALL
X MARCH
ROTATE
AND
VALUE
IF
J
EQ. 2
ESTABLISH
THE SIGN
OF V
SET OLD V = V


EXAM
COMPUTE THE
PRODUCT OF
OLD V AND V
1100
GT.O.D+0
IF
EXAM
LT.O.D+0
1200
1300
CALL
VALUE
OBTAIN ACTUAL
V NEW FOR THE
INTERPOLATED θ
COMPUTE
CHECK=V NEW * SIGN


IF
DABS(V)
LT.EPS
GT.EPS
1300
1300
REPLACE THE
COORDINATE
PAIR ASSOCIATED
WITH POSITIVE V
REINTERPOLATE
GT.O.D+0
IF
CHECK
GT.O.D+0
REPLACE THE
COORDINATE
PAIR ASSOCIATED
WITH NEGATIVE V
REINTERPOLATE
1600
COMPUTE THE
INTEGRAND
AND ITS
ABSOLUTE VALUE
SUM THE
INTEGRAND
AND ITS
ABSOLUTE VALUE


COMPUTE
TERM=DABS(ADD)
IF
TERM
LT.EPS
KOUNT
INCREMENT
COUNTER
1000
IF
LT.5
KOUNT
EQ. 5
WRITE OUTPUT
STOP THE TIMER
STOP

```
C        THIS PROGRAM COMPUTES THE INVERSE LAPLACE TRANSFORM OF A
C     FUNCTION USING THE STEEPEST DESCENT CONTOUR
      IMPLICIT REAL*8 (A-H,O-Z)
      CALL SETIME
      CALL ERRSET (207,256,-1,1,1,209)
C
C     ****************************************************************
C     *                                                              *
C     *  THIS SEGMENT OF THE PROGRAM NUMERICALLY INTEGRATES ALONG THE *
C     *BROMWICH CONTOUR UNTIL IT INTERSECTS WITH THE DESIRED V=0      *
C     *CONTOUR.  LINEAR INTERPOLATION IS USED TO REFINE THE LOCATION. *
C     *                                                              *
      TWOPI = 4.D+0*DARSIN(1.D+0)                    LEGEND
      T = TWOPI                                      2
      KKKKK = 0
      AL = 1.25D-1                                   2
      J = 1
      MMMMM = 0
      SUM = 0.D+0
      SUMA = 0.D+0
      AA = 3.D-1                                     2
      X = AA
      Y = 0.D+0
      YHOLD = 0.D+0
      DELY = 2.5D-3                                  2
      NREG = 3                                       2
      KSIGN = 0
      LEVEL = -1
      CALL VALUE (X,Y,T,AL,UHOLD,VHOLD,MMMMM)
  200 Y = Y+DELY
      CALL VALUE (X,Y,T,AL,U,V,MMMMM)
      J = J+1
      IF (J.EQ.2) SIGN = V/DABS(V)
      IF (J.GT.2) TEST = V*SIGN
      IF (TEST.LE.0.D+0) GO TO 400
  300 ADD = ((U+UHOLD)*(Y-YHOLD))/TWOPI
      SUM = SUM+ADD
      ADDA = DABS(ADD)
      SUMA = SUMA+ADDA
      IF (KKKKK.EQ.1) WRITE (6,1800) ADD,ADDA
      UHOLD = U
      VHOLD = V
      YHOLD = Y
      GO TO 200
  400 KSIGN = KSIGN+1
      LEVEL = -LEVEL
      IF (KSIGN.EQ.NREG) GO TO 500
      SIGN = -SIGN
      GO TO 300
  500 YONE = Y-DELY
      YTWO = Y
      VONE = VHOLD
      VTWO = V
  600 YNEW = (VONE*(YTWO-YONE)/(VONE-VTWO))+YONE
      CALL VALUE (X,YNEW,T,AL,UNEW,VNEW,MMMMM)
      IF (DABS(VNEW).LT.1.D-11) GO TO 900
      TEST = VNEW*SIGN
      IF (TEST.GT.0.D+0) GO TO 700
      YTWO = YNEW
      VTWO = VNEW
      GO TO 800
  700 YONE = YNEW
      VONE = VNEW
  800 CONTINUE
      GO TO 600
  900 ADD = ((UNEW+UHOLD)*(YNEW-YHOLD))/TWOPI
      ADDA = DABS(ADD)
      SUM = SUM+ADD
      SUMA = SUMA+ADDA
      IF (KKKKK.EQ.1) WRITE (6,1800) ADD,ADDA
      WRITE (6,2100) MMMMM
      WRITE (6,2000) SUM,SUMA
C     *                                                              *
C     ****************************************************************
```

```
C
C     $$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
C     $                                                                  $
C     $  THIS SEGMENT OF THE PROGRAM PERFORMS NUMERICAL INTEGRATION      $
C     $ALONG THE V = 0 CONTOUR.                          LEGEND          $
C     $                                                                  $
      KOUNT = 0
      NUMBER = 5                                            4
      EPS = 1.D-11                                          4
      THETA = 0.D+0
      Y = YNEW
 1000 J = 1
      XHOLD = X
      YHOLD = Y
      UHOLD = UNEW
      VHOLD = VNEW
      MMMMM = 0
 1100 J = J+1
      CALL XMARCH (XHOLD,YHOLD,THETA,X,Y)
      CALL VALUE (X,Y,T,AL,U,V,MMMMM)
      IF (J.EQ.2) SIGN = V/DABS(V)
      IF (J.GT.2) EXAM = V*SIGN
      IF (EXAM.LE.0.D+0) GO TO 1200
      VOLD = V
      THETAO = THETA
      CALL ROTATE (LEVEL,SIGN,THETA)
      GO TO 1100
 1200 THONE = THETAO
      THTWO = THETA
      VONE = VOLD
      VTWO = V
 1300 THNEW = (((THONE-THTWO)/(VTWO-VONE))*VONE)+THONE
      CALL XMARCH (XHOLD,YHOLD,THNEW,X,Y)
      CALL VALUE (X,Y,T,AL,UNEW,VNEW,MMMMM)
      IF (DABS(VNEW).LT.1.D-11) GO TO 1600
      CHECK = VNEW*SIGN
      IF (CHECK.GT.0.D+0) GO TO 1400
      THTWO = THNEW
      VTWO = VNEW
      GO TO 1500
 1400 THONE = THNEW
      VONE = VNEW
 1500 CONTINUE
      GO TO 1300
 1600 ADD = ((UNEW+UHOLD)*(Y-YHOLD)+(VNEW+VHOLD)*(X-XHOLD))/TWOPI
      ADDA = DABS(ADD)
      SUM = SUM+ADD
      SUMA = SUMA+ADDA
      IF (KKKKK.EQ.1) WRITE (6,1800) ADD,ADDA
C
C     ####################################################################
C     #                                                                  #
C     #  THIS IS THE TERMINATION CRITERION SEGMENT OF THE ALGORITHM.     #
C     #
      TERM = DABS(ADD)
      IF (TERM.LT.EPS) KOUNT=KOUNT+1
      IF (TERM.GE.EPS) KOUNT=0
      IF (KOUNT.EQ.NUMBER) GO TO 1700
C     #                                                                  #
C     ####################################################################
C
      THETA = THNEW
      EXAM = -EXAM
      GO TO 1000
 1700 WRITE (6,2000) SUM,SUMA
      WRITE (6,2100) MMMMM
      CALL GETIME (IET)
      EL = DFLOAT(IET)*2.6D-5
      WRITE (6,1900) EL
C     $                                                                  $
C     $$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
C
      STOP
C
```

```
C
 1800 FORMAT (1X,2E30.10,/)
 1900 FORMAT (1X,'ELAPSED TIME DURING THE COMPUTATION IS',E20.5)
 2000 FORMAT (1X,2E20.6)                                          LEGEND
 2100 FORMAT (1X,I8)
      END
      SUBROUTINE XMARCH (XHOLD,YHOLD,THETA,X,Y)
      IMPLICIT REAL*8 (A-H,O-Z)
      X = XHOLD-2.5D-3*DCOS(THETA)                                5
      Y = YHOLD+2.5D-3*DSIN(THETA)                                5
      RETURN
      END
      SUBROUTINE ROTATE (LEVEL,SIGN,THETA)
      IMPLICIT REAL*8 (A-H,O-Z)
      IF (LEVEL.NE.1) GO TO 100
      IF (SIGN.LT.0.D+0) THETA=THETA-5.D-2
      IF (SIGN.GT.0.D+0) THETA=THETA+5.D-2
      GO TO 200
  100 IF (SIGN.LT.0.D+0) THETA=THETA+5.D-2
      IF (SIGN.GT.0.D+0) THETA=THETA-5.D-2
  200 RETURN
      END
```

LEGEND

2. Mandatory user output.

4. May be altered by the user at his discretion.

5. Polar radius R = .025 built into program, may be changed by user.

## SECTION A-3. SHANKS' ACCELERATOR USER INSTRUCTIONS

The algorithm of this section is very similar to that of Section A-1. Only those features which differ have been discussed in this instruction.

1. Contour integration is performed using a simple parameterized curve for a finite number of points on this contour. This finite number of terms must be prescribed by the user. The integer variable M is used for this purpose.

2. The dimension statements at the beginning of the program must provide storage space in linear arrays which are equal to or larger than the value assigned to M. (These quantities will subsequently be used with Shanks' accelerator. Note particularly that the array is doubly indexed.)

3. The integer variable N specifies the number of iterates of Shanks' accelerator which are desired by the user. However, the assigned value of N cannot be greater than 6 as the program is now written.

4. A sample program output appears in Chapter VI of this thesis. Column 1 represents the natural result obtained after M evaluations of the addend, without application of the accelerator.

```
C     THIS PROGRAM CALCULATES THE INVERSE LAPLACE TRANSFORM OF A
C     FUNCTION USING A FINITE NUMBER OF TERMS ALONG A SIMPLE
C     PARAMETERIZED CONTOUR CF INTEGRATION WITH SHANKS' ACCELERATOR
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSICN A(10,100) , ACD(100)
      DIMENSICN XX(100),YY(100),UU(100),VV(100)
      CALL ERRSET (207,256,-1,1,1,209)
      MMMMM = 0
      TWOPI = 4.D+0*DARSIN(1.D+0)
      AL = 1.25D-1
      T = TWOPI
      AA = 3.D-1
      P = 0.D+0
      M = 20
      NN = 0
      N = 5
      CALL CURVE (AA,P,XO,YO)
      CALL VALUE (XO,YO,T,AL,UO,VO,MMMMM)
C
C     $$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
C     $                                                              $
C
C     $  THE FCLLOWING SEGMENT OF THE PROGRAM PERFORMS THE CONTOUR   $
C     $INTEGRATION FOR A FINITE NUMBER CF TERMS USING THE SIMPLE     $
C     $PARAMETERIZED CURVE.                                          $
C     $                                                              $
  200 NN = NN+1
      CALL INCREP (P)
      CALL CURVE (AA,P,X,Y)
      CALL VALUE (X,Y,T,AL,U,V,MMMMM)
      ACD(NN) = ((V+VO)*(X-XO)+(U+UO)*(Y-YO))/TWOPI
      XO = X
      YO = Y
      UC = U
      VO = V
      XX(NN) = X
      YY(NN) = Y
      UU(NN) = U
      VV(NN) = V
      IF (NN.EQ.M) GO TO 300
      GO TO 200
  300 CONTINUE
C     $                                                              $
C     $$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
C
C     ****************************************************************
C     *                                                              *
C     *  THIS SEGMENT OF THE PROGRAM IMPLEMENTS SHANKS' ACCELERATOR IN *
C     *ACCORDANCE WITH EQUATION 28 OF CHAPTER 6 OF THIS THESIS.      *
C     *                                                              *
      B = 1.D+0
      EPS = 1.D-12
C
C
      DC 400 I=1,N
C
C
      DC 400 J=1,M
  400 A(I,J) = 0.D+0
C
C
      I = 1
C
C
      DO 500 J=1,M
      JM = J-1
      B = ADD(J)
      IF (J.EQ.1) A(1,1) = B
      IF (J.GT.1) A(1,J) = B+A(1,JM)
  500 CCNTINUE
C
C
C
C
```

```
      DO 600 I=2,N
      IM = I-1
      K = 2*I-1
C
C
      DO 600 J=K,M
      JM = J-1
      JMM = J-2
      A(I,J) = 1.D+0
      DEN = A(IM,J)+A(IM,JMM)-2.D+0*A(IM,JM)
      IF (DABS(DEN).GT.EPS) A(I,J)=(A(IM,J)*A(IM,JMM)-A(IM,JM)**2)/DEN
  600 CONTINUE
C
C
C
C
      DO 700 J=2,M
  700 WRITE (6,800) J,(A(I,J),I=1,N)
C
C     *                                                                  *
C     ********************************************************************
C
      STOP
C
C
  800 FORMAT (5X,I3,6F18.14)
      END
      SUBROUTINE CURVE (A,P,X,Y)
      IMPLICIT REAL*8 (A-H,O-Z)
      X = A-(P*P)
      Y = P
      RETURN
      END
      SUBROUTINE INCREP (P)
      IMPLICIT REAL*8 (A-H,O-Z)
      P = P+1.D-1
      RETURN
      END
```

## SECTION A-4. SPECIAL CONTOUR ALGORITHM FOR CASES AS TREATED IN CHAPTER EIGHT USER INSTRUCTIONS

The integration contour used in this section is a vertical path to a specified value of y, followed by a simple parabola. The algorithm of this section is very similar to that of Section A-1. Only those items which differ are discussed in this section.

1. Contour integration is performed in the initial segment of the program along the Bromwich contour. The integer variable JJJJ specifies the number of poles of the function $f(s) = \frac{1}{s}\tanh(\frac{\alpha s}{2})$ above the real axis that are to be enclosed to the left of the distorted Bromwich contour, which is used in the second segment of the program.

2. The variable HALT computes the elevation above the real axis corresponding to the input value of JJJJ for the specified hyperbolic tangent.

3. The variable FLAG is internally assigned a value of zero or one, corresponding to the first or second segments of the contour of integration which the algorithm employs. No action is required by the user.

4. The integer variable KSTART is utilized in the outer loop of the program to increment the value of t for which F(t) is computed.

5. The linear arrays STORE and STASH are used to convert the values of F(t) and t to single precision, which is required for the graphics package associated with the IBM 360/67.

6. SUBROUTINE VALUE contains the real arithmetic coding of the function $g(s)$ for the hyperbolic tangent function of Chapter VIII of this thesis. Standard trigonometric identities for the hyperbolic functions of a complex argument have been utilized.

Note: This procedure is obviously keyed to the particular function $f(s) = \frac{1}{s}\tanh(\frac{\alpha s}{2})$. The user must modify it for other functions. One obvious modification which would generalize the procedure would simply be to specify rather than calculate the value of HALT, i.e., input HALT rather than JJJJJ.

```
C      THIS PROGRAM COMPUTES THE INVERSE LAPLACE TRANSFCRM OF A
C      FUNCTION USING THE SIMPLE PARAMETERIZED CONTOUR OF INTEGRATION
C      WITH MODIFICATION TO VARY THE ELEVATION FROM WHICH THE CURVE IS
C      LAUNCHED.  THE BROMWICH CONTOUR IS USED IN THE INITIAL SEGMENT.
       IMPLICIT REAL*8 (A-H,O-Z)
       REAL*4 STORE, STASH
       DIMENSION STORE(50), STASH(50)
       CALL ERRSET (207,256,-1,1,1,209)
       TWOPI = 4.D+0*DARSIN(1.D+0)
       AL = 1.D+1
       JJJJJ = 15
       KKKKK = 0
       NUMBER = 5
       KSTART = 50
C
       DO 400 KKK = 1, KSTART
       FLAG = 0.D+0
       HALT = (DFLOAT(JJJJJ)*TWOPI)/(2.D+0*AL)
       ACD = 0.D+0
       ACDA = 0.D+0
       SUM = 0.D+0
       SUMA = 0.D+0
       LOSC = 0
       MMMMM = 0
       AK = KKK
       AA = 3.D-1
       T = 5.D-1+AK*1.D+0
       P = 0.D+0
       KOUNT = 0
       NN = 0
       EPS = 1.D-11
       DELP = 1.D-1
       CALL CURVE (AA,P,HALT,FLAG,XO,YO)
       CALL VALUE (XO,YO,T,AL,UC,VO,MMMMM)
       TIME = C.D+0
       CALL SETIME
  200  NN = NN+1
       CALL INCREP (HALT,DELP,F,FLAG)
       CALL CURVE (AA,P,HALT,FLAG,X,Y)
       CALL VALUE (X,Y,T,AL,U,V,MMMMM)
       ADD = ((V+VO)*(X-XO)+(U+UO)*(Y-YO))/TWOPI
       IF (NN.EQ.1) ADDOLD=ADC
       PROD = ADD*ADDOLD
       IF (PROD.LT.0.D+0) LOSC=LOSC+1
       ADDA = DABS(ADD)
       SUM = SUM+ADD
       SUMA = SUMA+ADDA
       TEST = DABS(ADD)
       IF (TEST.LT.EPS) KOUNT=KCUNT+1
       IF (TEST.GE.EPS) KOUNT=0
       CALL GETIME (IET)
       EL = DFLOAT(IET)*2.6D-5
       TIME = TIME+EL
       CALL SETIME
       IF(KOUNT.EQ.NUMBER) GO TO 300
       XC = X
       YO = Y
       UO = U
       VC = V
       ADDOLD = ADD
       GO TO 200
  300  CCNTINUE
       WRITE (6,500) NN,TIME
       WRITE (6,600) T, SUM, SUMA, X, Y, LOSC, MMMMM
       STORE(KKK) = SNGL(T)
       STASH(KKK) = SNGL(SUM)
  400  CONTINUE
       CALL PLCTG(STORE,STASH,50,1,1,0,'T',1,'F(T)',4,0.0,55.0,-2.0,2.C,
      16.0,6.0)
       CALL PLOT(0.0,0.0,999)
C
       STOP
C
  500  FORMAT (1X,'TOTAL ELAPSED TIME AFTER ',I5,' ITERATIONS IS ',E20.5)
```

```
600 FORMAT (1X,1P5E20.5,2I8,/)
    END
    SUBROUTINE CURVE (AA,P,HALT,FLAG,X,Y)
    IMPLICIT REAL*8 (A-H,O-Z)
    IF (P.LT.HALT.AND.FLAG.NE.1.D+0) GO TO 100
    X = AA-((P-HALT)*(P-HALT))
    Y = P
    GO TO 200
100 X = AA
    Y = P
200 RETURN
    END
    SUBROUTINE INCREP (HALT,DELP,P,FLAG)
    IMPLICIT REAL*8 (A-H,O-Z)
    P = P+DELP
    IF (P.EQ.HALT) FLAG=1.D+0
    RETURN
    END
    SUBROUTINE VALUE (X,Y,T,AL,U,V,MMMMM)
    IMPLICIT REAL*8 (A-H,O-Z)
    MMMMM = MMMMM+1
    TEM1 = (AL*X)
    TEM2 = (AL*Y)
    TEM3 = DSINH(TEM1)
    TEM4 = DSIN(TEM2)
    TEM5 = DCOSH(TEM1)+DCOS(TEM2)
    TEM6 = TEM3/TEM5
    TEM7 = TEM4/TEM5
    TEM8 = (TEM6*X)+(TEM7*Y)
    TEM9 = (TEM7*X)-(TEM6*Y)
    TEM10 = (X*X)+(Y*Y)
    TEM11 = TEM8/TEM10
    TEM12 = TEM9/TEM10
    TEM13 = DEXP(X*T)*DCOS(Y*T)
    TEM14 = DEXP(X*T)*DSIN(Y*T)
    U = (TEM13*TEM11)-(TEM14*TEM12)
    V = (TEM14*TEM11)+(TEM13*TEM12)
    RETURN
    END
```

## APPENDIX B

## LISTING OF TEST CASES AND EXPERIMENTAL RESULTS

The Laplace transformation pairs f(s) and F(t) which we have used as test cases are listed within this appendix. Also included for each function f(s) is the listing of the SUBROUTINE VALUE and a typical result of numerical inversion at one specific value of t. Case 9 also shows a plot of F(t) versus t obtained numerically.

The results obtained by numerical inversion have been compared where applicable, with tabulated values listed in (1). In all other cases the value of F(t) has been calculated on the IBM 360/67.

The symbols which are used in the presentation of results are defined below:

AL - constant associated with f(s)

BE - constant associated with f(s)

T - time for which F(t) is evaluated

AA - real axis starting position of the integration contour

$x_f$ - final value of x at termination

$y_f$ - final value of y at termination

MMMMM - total number of calls made to SUBROUTINE VALUE

LOSC - number of sign changes between successive addends

SUM - value of F(t) (equation (16))

SUMA - value of G(t) (equation (19))

TIME - clock time (sec) required for numerical integration

ANALYTICAL - analytical value of F(t)

In all cases we used N = 5 and EPS = 1.D-11.

| f(s) | Case No. | f(s) | Case No. |
|---|---|---|---|
| $1/s$ | 1 | $s/(s^2+a^2)^2$ | 19 |
| $1/s^2$ | 2 | $s^2/(s^2+a^2)^2$ | 20 |
| $1/s^n$ | 3 | $s^3/(s^2+a^2)^2$ | 21 |
| $1/s^n$ | 4 | $(s^2-a^2)/(s^2+a^2)^2$ | 22 |
| $1/(s-a)$ | 5 | $1/(s^2-a^2)^2$ | 23 |
| $1/(s-a)^n$ | 6 | $s/(s^2-a^2)^2$ | 24 |
| $1/(s-a)^n$ | 7 | $s^2/(s^2-a^2)^2$ | 25 |
| $1/(s^2+a^2)$ | 8 | $s^3/(s^2-a^2)^2$ | 26 |
| $s/(s^2+a^2)$ | 9 | $(s^2+a^2)/(s^2-a^2)^2$ | 27 |
| $1/((s-b)^2+a^2)$ | 10 | $1/(s^2+a^2)^3$ | 28 |
| $(s-b)/((s-b)^2+a^2)$ | 11 | $s/(s^2+a^2)^3$ | 29 |
| $1/(s^2-a^2)$ | 12 | $s^2/(s^2+a^2)^3$ | 30 |
| $s/(s^2-a^2)$ | 13 | $s^3/(s^2+a^2)^3$ | 31 |
| $1/((s-b)^2-a^2)$ | 14 | $s^4/(s^2+a^2)^3$ | 32 |
| $(s-b)/((s-b)^2-a^2)$ | 15 | $s^5/(s^2+a^2)^3$ | 33 |
| $1/(s-b)(s-a)$ | 16 | $(3s^2-a^2)/(s^2+a^2)^3$ | 34 |
| $s/(s-a)(s-b)$ | 17 | $(s^3-3a^2s)/(s^2+a^2)^3$ | 35 |
| $1/(s^2+a^2)^2$ | 18 | $\frac{s^4-6a^2s^2+a^4}{(s^2+a^2)^4}$ | 36 |

| f(s) | Case No. | f(s) | Case No. |
|---|---|---|---|
| $\dfrac{s^3-a^2s}{(s^2+a^2)^4}$ | 37 | $s/(s^3-a^3)$ | 52 |
| $1/(s^2-a^2)^3$ | 38 | $s^2/(s^3-a^3)$ | 53 |
| $s/(s^2-a^2)^3$ | 39 | $1/(s^4-4a^4)$ | 54 |
| $s^2/(s^2-a^2)^3$ | 40 | $s/(s^4-4a^4)$ | 55 |
| $s^3/(s^2-a^2)^3$ | 41 | $s^2/(s^4+4a^4)$ | 56 |
| $s^4/(s^2-a^2)^3$ | 42 | $s^3/(s^4+4a^4)$ | 57 |
| $s^5/(s^2-a^2)^3$ | 43 | $1/(s^4-a^4)$ | 58 |
| $\dfrac{3s^2+a^2}{(s^2-a^2)^3}$ | 44 | $s/(s^4-a^4)$ | 59 |
| $\dfrac{s^3+3a^2s}{(s^2-a^2)^3}$ | 45 | $s^2/(s^4-a^4)$ | 60 |
| $\dfrac{s^4+6a^2s^2+a^4}{(s^2+a^2)^4}$ | 46 | $s^3/(s^4-a^4)$ | 61 |
| $\dfrac{s^3+a^2s}{(s^2-a^2)^4}$ | 47 | $\dfrac{1}{\sqrt{s+a}\ \sqrt{s+b}}$ | 62 |
| $1/(s^3+a^3)$ | 48 | $\dfrac{1}{\sqrt{s}\ (s+a)}$ | 63 |
| $s/(s^3+a^3)$ | 49 | $\dfrac{1}{\sqrt{s}\ (s-a)}$ | 64 |
| $s^2/(s^3+a^3)$ | 50 | $\dfrac{1}{\sqrt{s-a}+b}$ | 65 |
| $1/(s^3-a^3)$ | 51 | | |

| f(s) | Case No. | f(s) | Case No. |
|---|---|---|---|
| $\frac{1}{\sqrt{s^2+a^2}}$ | 66 | $\frac{e^{-as}}{\sqrt{s}}$ | 79 |
| $\frac{1}{\sqrt{s^2-a^2}}$ | 67 | $\frac{e^{-as}}{s^{n+1}}$ | 80 |
| $\frac{(\sqrt{s^2+a^2} - s)^n}{\sqrt{s^2+a^2}}$ | 68 | $\frac{e^{-a\sqrt{s}}}{\sqrt{s}}$ | 81 |
| $\frac{(s - \sqrt{s^2-a^2})^n}{\sqrt{s^2-a^2}}$ | 69 | $e^{-a\sqrt{s}}$ | 82 |
| $\frac{e^{b(s-\sqrt{s^2+a^2}}}{\sqrt{s^2+a^2}}$ | 70 | $\frac{1-e^{-a\sqrt{s}}}{s}$ | 83 |
| $\frac{e^{-b\sqrt{s^2+a^2}}}{\sqrt{s^2+a^2}}$ | 71 | $\frac{e^{-a\sqrt{s}}}{s}$ | 84 |
| $1/(s^2+a^2)^{3/2}$ | 72 | $\frac{e^{-a\sqrt{s}}}{\sqrt{s}(\sqrt{s} + b}$ | 85 |
| $s/(s^2+a^2)^{3/2}$ | 73 | $Ln(\frac{s+a}{s+b})$ | 86 |
| $s^2/(s^2+a^2)^{3/2}$ | 74 | | |
| $1/(s^2-a^2)^{3/2}$ | 75 | $\frac{Ln(\frac{s^2+a^2}{a^2})}{2s}$ | 87 |
| $s/(s^2-a^2)^{3/2}$ | 76 | | |
| $s^2/(s^2-a^2)^{3/2}$ | 77 | $\frac{Ln(\frac{s+a}{a})}{s}$ | 88 |
| $\frac{e^{-a/s}}{\sqrt{s}}$ | 78 | $\frac{Ln(s)}{s}$ | 89 |

| $f(s)$ | Case No. |
| --- | --- |
| $\frac{Ln^2(s)}{s}$ | 90 |
| $\frac{Ln\left(\frac{S+\sqrt{s^2+a^2}}{a}\right)}{\sqrt{s^2+a^2}}$ | 91 |
| $\frac{Ln\left(\frac{\sqrt{S^2+a^2}+a}{s}\right)}{\sqrt{s^2+a^2}}$ | 92 |
| $\frac{e^{-as}}{s}$ | 93 |
| $\frac{e^{-a^2/4s}}{s}$ | 94 |
| $\frac{1}{s}\left(\frac{s-a}{s+a}\right)^2$ | 95 |

Case 1.

$$\mathcal{L}^{-1}\left[\frac{1}{s}\right] = 1$$

```
      SUBROUTINE VALUE (X,Y,T,AL,U,V,MMMMM)
C     THIS IS TABLE ENTRY NUMBER 01 OF LAPLACE TRANSFORMS BY SPIEGEL
      IMPLICIT REAL*8 (A,B,D-H,O-Z)
      IMPLICIT COMPLEX*16 (C)
      MMMMM = MMMMM + 1
      XPTR = X*T
      XPTI = Y*T
      CXPT = DCMPLX(XPTR,XPTI)
      CNUM = CDEXP(CXPT)
      CDENOM = DCMPLX(X,Y)
      CRESLT = CNUM/CDENOM
      U = DREAL(CRESLT)
      V = DIMAG(CRESLT)
      RETURN
      END
```

AL = 1.25D-1

T = 8.D+0

AA = 0.2

$X_F$ = -4.64

$y_F$ = 2.2

MMMMM = 23

LOSC = 5

SUM = 1.00000D+0

SUMA = 1.42588D+0

TIME = 0.79794D-1

ANALYTICAL = 1.00000D+0

Case 2.

$$\mathcal{L}^{-1}\left[\frac{1}{s^2}\right] = t$$

```
      SUBROUTINE VALUE (X,Y,T,AL,U,V,MMMMM)
C     THIS IS TABLE ENTRY NUMBER 02 OF LAPLACE TRANSFORMS BY SPIEGEL
      IMPLICIT REAL*8 (A,B,D-H,O-Z)
      IMPLICIT COMPLEX*16 (C)
      MMMMM = MMMMM + 1
      CS1 = DCMPLX(X,Y)
      XPTR = X*T
      XPTI = Y*T
      CXPT = DCMPLX(XPTR,XPTI)
      CNUM = CDEXP(CXPT)
      CDENOM = DCMPLX(X,Y)
      CDENOM = CS1*CS1
      CRESLT = CNUM/CDENOM
      U = DREAL(CRESLT)
      V = DIMAG(CRESLT)
      RETURN
      END
```

AL = 1.25D-1

T = 8.D+0

AA = 0.2

$x_F$ = -4.64

$y_F$ = 2.2

MMMMM = 23

LOSC = 4

SUM = 8.00000D+0

SUMA = 8.04195D+0

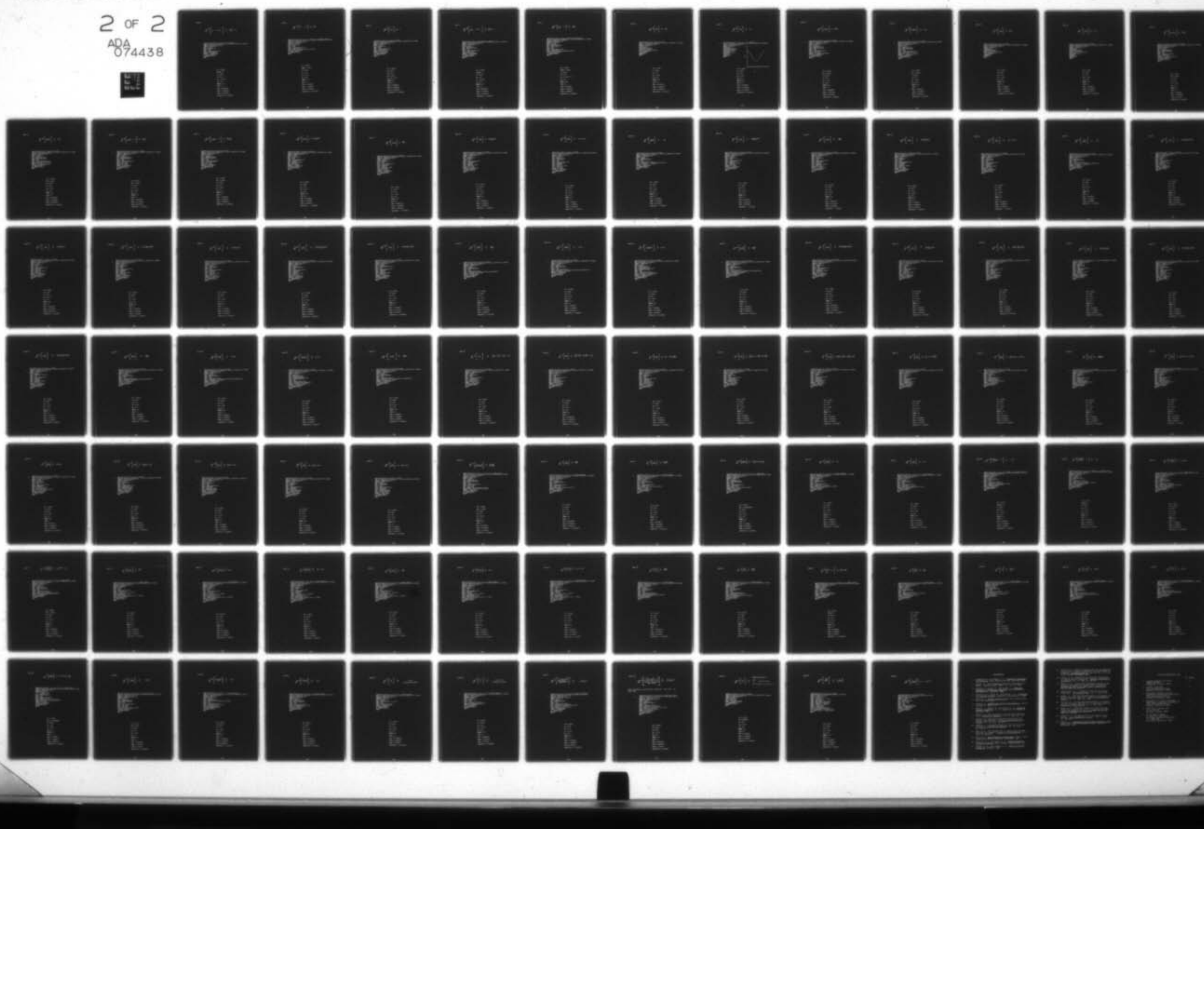
TIME = 0.53248D-1

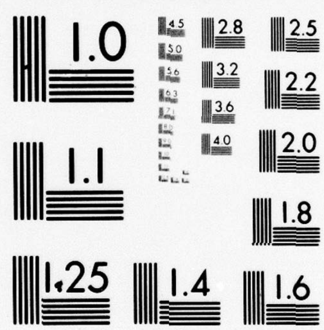
ANALYTICAL = 8.00000D+0

AD-A074 438 NAVAL POSTGRADUATE SCHOOL MONTEREY CA F/G 12/1
NUMERICAL INVERSION OF LAPLACE TRANSFORMS.(U)
JUN 79 J H DUNCAN

UNCLASSIFIED NL

2 OF 2
ADA 074438






MICROCOPY RESOLUTION TEST CHART

NATIONAL BUREAU OF

Case 3.

$$\mathcal{L}^{-1}\left[\frac{1}{s^n} \qquad n = 1,2,3,\ldots\right] = \frac{t^{n-1}}{(n-1)!}, \quad 0! = 1$$

```
      SUBROUTINE VALUE(X,Y,T,AL,NPOWER,U,V,MMMMM)
C     THIS IS TABLE ENTRY NUMBER 03 OF LAPLACE TRANSFORMS BY SPIEGEL
      IMPLICIT REAL*8 (A,B,D-H,O-Z)
      IMPLICIT COMPLEX*16 (C)
      MMMMM = MMMMM + 1
      ZERO = 0.D+0
      CS1 = DCMPLX(X,Y)
      XPTR = X*T
      XPTI = Y*T
      CXPT = DCMPLX(XPTR,XPTI)
      CNUM = CDEXP(CXPT)
      CDENOM = CS1**NPOWER
      CRESLT = CNUM/CDENOM
      U = DREAL(CRESLT)
      V = DIMAG(CRESLT)
      RETURN
      END
```

AL = 1.25D-1

NPOWER = 3

T = 8.D+0

AA = 0.3

$x_p$ = -4.54

$y_p$ = 2.2

MMMMM = 23

LOSC = 3

SUM = 3.20000D+1

SUMA = 3.20008D+1

TIME = 0.83694D-1

ANALYTICAL = 3.20000D+1

Case 4.

$$\mathcal{L}^{-1}\left[\frac{1}{s^n} \quad n > 0\right] = \frac{t^{n-1}}{\Gamma(n)}$$

```
      SUBROUTINE VALUE (X, Y, T, AL, POWER, U, V, MMMMM, NCALL)
C     THIS IS TABLE ENTRY NUMBER 04 OF LAPLACE TRANSFORMS BY SPIEGEL
      IMPLICIT REAL*8 (A,B,D-H,O-Z)
      IMPLICIT COMPLEX*16 (C)
      DIMENSION NCALL(5)
      MMMMM = MMMMM + 1
      CS1 = DCMPLX(X,Y)
      XPTR = X*T
      XPTI = Y*T
      CXPT = DCMPLX(XPTR,XPTI)
      CNUM = CDEXP(CXPT)
      CALL CPOWER (CS1,CDENOM,POWER,1,NCALL)
      CRESLT = CNUM/CDENOM
      U = DREAL(CRESLT)
      V = DIMAG(CRESLT)
      RETURN
      END
```

AL = 1.25D-1

POWER = 1.7D+0

T = 8.D+0

AA = 0.2

$x_F$ = -4.64

$y_F$ = 2.2

MMMMM = 23

LOSC = 4

SUM = 4.71815D+0

SUMA = 4.8324D+0

TIME = 0.85098D-1

ANALYTICAL = 4.71815D+0

Case 5.

$$\mathcal{L}^{-1}\left[\frac{1}{s-a}\right] = e^{at}$$

```
      SUBROUTINE VALUE (X,Y,T,AL,U,V,MMMMM)
C     THIS IS TABLE ENTRY NUMBER 05 OF LAPLACE TRANSFORMS BY SPIEGEL
      IMPLICIT REAL*8 (A,B,D-H,O-Z)
      IMPLICIT COMPLEX*16 (C)
      MMMMM = MMMMM + 1
      ZERO = 0.D+0
      CS1 = DCMPLX(X,Y)
      XPTR = X*T
      XPTI = Y*T
      CXPT = DCMPLX(XPTR,XPTI)
      CNUM = CDEXP(CXPT)
      CS2 = DCMPLX(AL,ZERO)
      CDENOM = CS1 - CS2
      CRESLT = CNUM/CDENOM
      U = DREAL(CRESLT)
      V = DIMAG(CRESLT)
      RETURN
      END
```

AL = 1.25D-1

T = 8.D+0

AA = 0.3

$X_F$ = -4.99

$y_F$ = 2.3 2.3

MMMMM = 23

LOSC = 5

SUM = 2.71828D+0

SUMA = 3.59474D+0

TIME = 0.89050D+0

ANALYTICAL = 2.71828D+0

Case 6.

$$\mathcal{L}^{-1}\left[\frac{1}{(s-a)^n} \quad n = 1, 2, 3, \ldots\right] = \frac{t^{n-1} e^{at}}{(n-1)!}, \quad 0! = 1$$

```
      SUBROUTINE VALUE(X,Y,T,AL,NPOWER,U,V,MMMMM)
C     THIS IS TABLE ENTRY NUMBER 06 OF LAPLACE TRANSFORMS BY SPIEGEL
      IMPLICIT REAL*8 (A,B,D-H,O-Z)
      IMPLICIT COMPLEX*16 (C)
      MMMMM = MMMMM + 1
      ZERO = 0.D+0
      CS1 = DCMPLX(X,Y)
      CS2 = DCMPLX(AL,ZERO)
      XPTR = X*T
      XPTI = Y*T
      CXPT = DCMPLX(XPTR,XPTI)
      CNUM = CDEXP(CXPT)
      CS3 = CS1 - CS2
      CDENOM = CS3**NPOWER
      CRESLT = CNUM/CDENOM
      U = DREAL(CRESLT)
      V = DIMAG(CRESLT)
      RETURN
      END
```

AL = 1.25D-1

NPOWER = 3

T = 8.D+0

AA = 0.4

$X_F$ = -4.44

$y_F$ = 2.2

MMMMM = 23

LOSC = 3

SUM = 8.69850D+1

SUMA = 8.69865D+1

TIME = 0.75790D-1

ANALYTICAL = 8.69850D+1

Case 7.

$$\mathcal{L}^{-1}\left[\frac{1}{(s-a)^n} \quad n > 0\right] = \frac{t^{n-1}e^{at}}{\Gamma(n)}$$

```
      SUBROUTINE VALUE (X,Y,T,AL,NPOWER,U,V,MMMMM,NCALL)
C     THIS IS TABLE ENTRY NUMBER 07 OF LAPLACE TRANSFORMS BY SPIEGEL
      IMPLICIT REAL*8 (A,B,D-H,O-Z)
      IMPLICIT COMPLEX*16 (C)
      DIMENSION NCALL(5)
      MMMMM = MMMMM + 1
      ZERO = 0.D+0
      CS1 = DCMPLX(X,Y)
      CS2 = DCMPLX(AL,ZERO)
      CS3 = CS1 - CS2
      XPTR = X*T
      XPTI = Y*T
      CXPT = DCMPLX(XPTR,XPTI)
      CNUM = CDEXP(CXPT)
      CALL CPOWER (CS3,CDENOM,POWER,1,NCALL)
      CRESLT = CNUM/CDENOM
      U = DREAL(CRESLT)
      V = DIMAG(CRESLT)
      RETURN
      END
```

AL = 1.25D-1

POWER = 1.7D+0

T = 8.D+0

AA = 0.3

$x_p$ = -4.54 -4.54

$y_p$ = 2.2

MMMMM = 23

LOSC = 4

SUM = 1.28253D+1

SUMA = 1.30533D+1

TIME = 0.96928D-1

ANALYTICAL = 1.28253D+1

Case 8.

$$\mathcal{L}^{-1}\left[\frac{1}{s^2+a^2}\right] = \frac{\sin at}{a}$$

```
      SUBROUTINE VALUE(X,Y,T,AL,U,V,MMMMM)
C     THIS IS TABLE ENTRY NUMBER 08 OF LAPLACE TRANSFORMS BY SPIEGEL
      IMPLICIT REAL*8 (A,B,D-H,O-Z)
      IMPLICIT COMPLEX*16 (C)
      MMMMM = MMMMM + 1
      ZERO=0.D+0
      CS=DCMPLX(X,Y)
      CAL=DCMPLX(AL,ZERO)
      CT=DCMPLX(T,ZERO)
      CDEN = ((CS*CS)+(CAL*CAL))
      CEXP=CDEXP(CS*CT)
      C = CEXP/CDEN
      U=DREAL(C)
      V=DIMAG(C)
      RETURN
      END
```

AL = 1.25D-1

T = 8.D+0

AA = 0.3

$X_F$ = -4.54

$y_F$ = 2.2

MMMMM = 23

LOSC = 4

SUM = 6.73177D+0

SUMA = 7.03097D+0

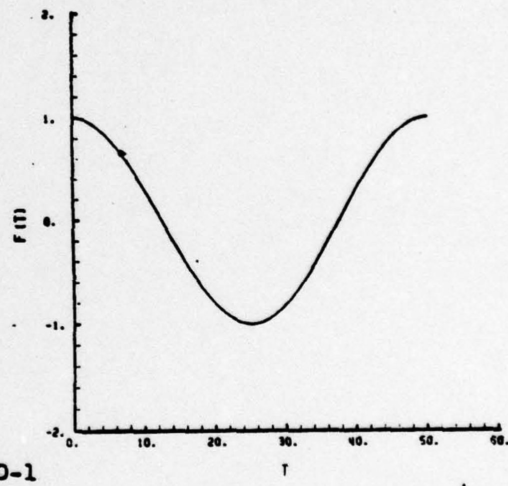
TIME = 0.86294D-1

ANALYTICAL = 6.73177D+0

Case 9.

$$\mathcal{L}^{-1}\left[\frac{s}{s^2+a^2}\right] = \cos at$$

```
      SUBROUTINE VALUE (X, Y, T, AL, U, V, MMMMM)
C     THIS IS TABLE ENTRY NUMBER 09 OF LAPLACE TRANSFORMS BY SPIEGEL
      IMPLICIT REAL*8 (A,B,D-H,O-Z)
      IMPLICIT COMPLEX*16 (C)
      MMMMM = MMMMM + 1
      ZERO=0.D+0
      CS=DCMPLX(X,Y)
      CAL=DCMPLX(AL,ZERO)
      CT=DCMPLX(T,ZERO)
      CDEN=CS**2+CAL**2
      C=CS/CDEN
      CEXP=CDEXP(CS*CT)
      C=C*CEXP
      U=DREAL(C)
      V=DIMAG(C)
      RETURN
      END
```




AL = 1.25D-1

T = 8.D+0

AA = 0.3

$X_F$ = -4.99

$y_F$ = 2.3

MMMMM = 24

LOSC = 5

SUM = 5.40302D-1

SUMA = 2.00010D+0

TIME = 0.11138D+0

ANALYTICAL = 5.40302D-1

Case 10.

$$\mathcal{L}^{-1}\left[\frac{1}{(s-b)^2+a^2}\right] = \frac{e^{bt}\sin at}{a}$$

```
      SUBROUTINE VALUE(X,Y,T,AL,BE,U,V,MMMMM)
C     THIS IS TABLE ENTRY NUMBER 10 OF LAPLACE TRANSFORMS BY SPIEGEL
      IMPLICIT REAL*8 (A,B,D-H,O-Z)
      IMPLICIT COMPLEX*16 (C)
      MMMMM = MMMMM + 1
      ZERO = 0.D+0
      CS1 = DCMPLX(X,Y)
      CS2 = DCMPLX(BE,ZERO)
      CS3 = DCMPLX(AL,ZERO)
      XPTR = X*T
      XPTI = Y*T
      CXPT = DCMPLX(XPTR,XPTI)
      CNUM = CDEXP(CXPT)
      CS4 = CS1 - CS2
      CDENOM = (CS4*CS4)+(CS3*CS3)
      CRESLT = CNUM/CDENOM
      U = DREAL(CRESLT)
      V = DIMAG(CRESLT)
      RETURN
      END
```

AL = 1.25D-1

BE = 1.25D-1

T = 8.D+0

AA = 0.3

$X_P$ = -4.54

$y_P$ = 2.2

MMMMM = 23

LOSC = 4

SUM = 1.82988D+1

SUMA = 1.88117D+1

TIME = 0.70044D-1

ANALYTICAL = 1.82988D+1

Case 11.

$$\mathcal{L}^{-1}\left[\frac{s-b}{(s-b)^2+a^2}\right] = e^{bt}\cos at$$

```
      SUBROUTINE VALUE(X,Y,T,AL,BE,U,V,MMMMM)
C     THIS IS TABLE ENTRY NUMBER 11 OF LAPLACE TRANSFORMS BY SPIEGEL
      IMPLICIT REAL*8 (A,B,D-H,O-Z)
      IMPLICIT COMPLEX*16 (C)
      MMMMM = MMMMM + 1
      ZERO = 0.D+0
      CS1 = DCMPLX(X,Y)
      CS2 = DCMPLX(BE,ZERO)
      CS3 = DCMPLX(AL,ZERO)
      XPTR = X*T
      XPTI = Y*T
      CXPT = DCMPLX(XPTR,XPTI)
      CNUM = (CS1-CS2)*CDEXP(CXPT)
      CS4 = CS1 - CS2
      CDENOM = (CS4*CS4)+(CS3*CS3)
      CRESLT = CNUM/CDENOM
      U = DREAL(CRESLT)
      V = DIMAG(CRESLT)
      RETURN
      END
```

AL = 1.25D-1

BE = 1.25D-1

T = 8.D+0

AA = 0.4

$X_F$ = -4.89

$y_F$ = 2.3

MMMMM = 24

LOSC = 5

SUM = 1.46869D+0

SUMA = 4.61766D+0

TIME = 0.78130D-1

ANALYTICAL = 1.46869D+0

Case 12.

$$\mathcal{L}^{-1}\left[\frac{1}{s^2 - a^2}\right] = \frac{\sinh at}{a}$$

```
      SUBROUTINE VALUE(X,Y,T,AL,U,V,MMMMM)
C     THIS IS TABLE ENTRY NUMBER 12 OF LAPLACE TRANSFORMS BY SPIEGEL
      IMPLICIT REAL*8 (A,B,D-H,O-Z)
      IMPLICIT COMPLEX*16 (C)
      MMMMM = MMMMM + 1
      ZERO=0.D+0
      CS=DCMPLX(X,Y)
      CAL=DCMPLX(AL,ZERO)
      CT=DCMPLX(T,ZERO)
      CDEN = ((CS*CS)-(CAL*CAL))
      CEXP=CDEXP(CS*CT)
      C = CEXP/CDEN
      U=DREAL(C)
      V=DIMAG(C)
      RETURN
      END
```

AL = 1.25D-1

T = 8.D+0

AA = 0.3

$X_P$ = -4.54

$y_P$ = 2.2

MMMMM = 23

LOSC = 4

SUM = 9.40162D+0

SUMA = 9.64889D+0

TIME = 0.10691D+0

ANALYTICAL = 9.40162D+0

Case 13.

$$\mathcal{L}^{-1}\left[\frac{s}{s^2 - a^2}\right] = \cosh at$$

```
      SUBROUTINE VALUE(X,Y,T,AL,U,V,MMMMM)
C     THIS IS TABLE ENTRY NUMBER 13 OF LAPLACE TRANSFORMS BY SPIEGEL
      IMPLICIT REAL*8 (A,B,D-H,O-Z)
      IMPLICIT COMPLEX*16 (C)
      MMMMM = MMMMM + 1
      ZERO=0.D+0
      CS=DCMPLX(X,Y)
      CAL=DCMPLX(AL,ZERO)
      CT=DCMPLX(T,ZERO)
      CDEN = ((CS*CS)-(CAL*CAL))
      C=CS/CDEN
      CEXP=CDEXP(CS*CT)
      C=C*CEXP
      U=DREAL(C)
      V=DIMAG(C)
      RETURN
      END
```

AL = 1.25D-1

T = 8.D+0

AA = 0.3

$X_F$ = -4.99

$y_F$ = 2.3

MMMMM = 24

LOSC = 5

SUM = 1.54308D+0

SUMA = 2.52401D+0

TIME = 0.85176D-1

ANALYTICAL = 1.54308D+0

Case 14.

$$\mathcal{L}^{-1}\left[\frac{1}{(s-b)^2-a^2}\right] = \frac{e^{bt}\sinh at}{a}$$

```
      SUBROUTINE VALUE(X,Y,T,AL,BE,U,V,MMMMM)
C     THIS IS TABLE ENTRY NUMBER 14 OF LAPLACE TRANSFORMS BY SPIEGEL
      IMPLICIT REAL*8 (A,B,D-H,O-Z)
      IMPLICIT COMPLEX*16 (C)
      MMMMM = MMMMM + 1
      ZERO = 0.D+0
      CS1 = DCMPLX(X,Y)
      CS2 = DCMPLX(BE,ZERO)
      CS3 = DCMPLX(AL,ZERO)
      XPTR = X*T
      XPTI = Y*T
      CXPT = DCMPLX(XPTR,XPTI)
      CNUM = CDEXP(CXPT)
      CS4 = CS1 - CS2
      CDENOM = (CS4*CS4)-(CS3*CS3)
      CRESLT = CNUM/CDENOM
      U = DREAL(CRESLT)
      V = DIMAG(CRESLT)
      RETURN
      END
```

AL = 1.25D-1

BE = 1.25D-1

T = 8.D+0

AA = 0.5

$X_F$ = -4.79

$y_F$ = 2.3

MMMMM = 24

LOSC = 4

SUM = 2.55562D+1

SUMA = 2.81818D+1

TIME = 0.74932D-1

ANALYTICAL = 2.55562D+1

Case 15.

$$\mathcal{L}^{-1}\left[\frac{s-b}{(s-b)^2-a^2}\right] = e^{bt}\cosh at$$

```
      SUBROUTINE VALUE(X,Y,T,AL,BE,U,V,MMMMM)
C     THIS IS TABLE ENTRY NUMBER 15 OF LAPLACE TRANSFORMS BY SPIEGEL
      IMPLICIT REAL*8 (A,B,D-H,O-Z)
      IMPLICIT COMPLEX*16 (C)
      MMMMM = MMMMM + 1
      ZERO = 0.D+0
      CS1 = DCMPLX(X,Y)
      CS2 = DCMPLX(BE,ZERO)
      CS3 = DCMPLX(AL,ZERO)
      XPTR = X*T
      XPTI = Y*T
      CXPT = DCMPLX(XPTR,XPTI)
      CNUM = (CS1-CS2)*CDEXP(CXPT)
      CS4 = CS1 - CS2
      CDENOM = (CS4*CS4)-(CS3*CS3)
      CRESLT = CNUM/CDENOM
      U = DREAL(CRESLT)
      V = DIMAG(CRESLT)
      RETURN
      END
```

AL = 1.25D-1

BE = 1.25D-1

T = 8.D+0

AA = 0.5

$X_F$ = -4.79

$y_F$ = 2.3

MMMMM = 24

LOSC = 5

SUM = 4.19453D+0

SUMA = 9.63599D+0

TIME = 0.77740D-1

ANALYTICAL = 4.19453D+0

Case 16.

$$\mathcal{L}^{-1}\left[\frac{1}{(s-a)(s-b)} \quad a \neq b\right] = \frac{e^{bt}-e^{at}}{b-a}$$

```
      SUBROUTINE VALUE(X,Y,T,AL,BE,U,V,MMMMM)
C     THIS IS TABLE ENTRY NUMBER 16 OF LAPLACE TRANSFORMS BY SPIEGEL
      IMPLICIT REAL*8 (A,B,D-H,O-Z)
      IMPLICIT COMPLEX*16 (C)
      MMMMM = MMMMM + 1
      ZERO = 0.D+0
      CS1 = DCMPLX(X,Y)
      CS2 = DCMPLX(BE,ZERO)
      CS3 = DCMPLX(AL,ZERO)
      XPTR = X*T
      XPTI = Y*T
      CXPT = DCMPLX(XPTR,XPTI)
      CNUM = CDEXP(CXPT)
      CS4 = CS1 - CS3
      CS5 = CS1 - CS2
      CDENOM = CS4*CS5
      CRESLT = CNUM/CDENOM
      U = DREAL(CRESLT)
      V = DIMAG(CRESLT)
      RETURN
      END
```

AL = 1.25D-1

BE = -1.25D-1

T = 8.D+0

AA = 0.4

$X_F$ = -4.89

$y_F$ = 2.3

MMMMM = 24

LOSC = 4

SUM = 9.40161D+0

SUMA = 1.07718D+1

TIME = 0.76648D-1

ANALYTICAL = 9.40161D+0

Case 17.

$$\mathcal{L}^{-1}\left[\frac{s}{(s-a)(s-b)} \quad a \neq b\right] = \frac{be^{bt} - ae^{at}}{b-a}$$

```
      SUBROUTINE VALUE(X,Y,T,AL,BE,U,V,MMMMM)
C     THIS IS TABLE ENTRY NUMBER 17 OF LAPLACE TRANSFORMS BY SPIEGEL
      IMPLICIT REAL*8 (A,B,D-H,O-Z)
      IMPLICIT COMPLEX*16 (C)
      MMMMM = MMMMM + 1
      ZERO = 0.D+0
      CS1 = DCMPLX(X,Y)
      CS2 = DCMPLX(BE,ZERO)
      CS3 = DCMPLX(AL,ZERO)
      XPTR = X*T
      XPTI = Y*T
      CXPT = DCMPLX(XPTR,XPTI)
      CNUM = CS1*CDEXP(CXPT)
      CS4 = CS1 - CS3
      CS5 = CS1 - CS2
      CDENOM = CS4*CS5
      CRESLT = CNUM/CDENOM
      U = DREAL(CRESLT)
      V = DIMAG(CRESLT)
      RETURN
      END
```

AL = 1.25D-1

BE = -1.25D-1

T = 8.D+0

AA = 0.3

$X_P$ = -4.99

$y_F$ = 2.3

MMMMM = 24

LOSC = 5

SUM = 1.54308D+0

SUMA = 2.52401D+0

TIME = 0.78988D-1

ANALYTICAL = 1.54308D+0

Case 18.

$$\mathcal{L}^{-1}\left[\frac{1}{(s^2+a^2)^2}\right] = \frac{\sin at - at\cos at}{2a^3}$$

```
      SUBROUTINE VALUE (X,Y,T,AL,U,V,MMMMM)
C     THIS IS TABLE ENTRY NUMBER 18 OF LAPLACE TRANSFORMS BY SPIEGEL
      IMPLICIT REAL*8 (A,B,D-H,O-Z)
      IMPLICIT COMPLEX*16 (C)
      MMMMM = MMMMM + 1
      ZERO = 0.D+0
      CS1 = DCMPLX(X,Y)
      XPTR = X*T
      XPTI = Y*T
      CXPT = DCMPLX(XPTR,XPTI)
      CNUM = CDEXP(CXPT)
      CS4 = CS1*CS1
      CS5 = DCMPLX(AL,ZERO)
      CS6 = CS5*CS5
      CS7 = CS4 + CS6
      CDENOM = CS7*CS7
      CRESLT = CNUM/CDENOM
      U = DREAL(CRESLT)
      V = DIMAG(CRESLT)
      RETURN
      END
```

AL = 1.25D-1

T = 8.D+0

AA = 0.3

$X_P$ = -4.11

$y_P$ = 2.1

MMMMM = 22

LOSC = 2

SUM = 7.70992D+1

SUMA = 7.70992D1

TIME = 0.51870D-1

ANALYTICAL 7.70992D+1

Case 19.

$$\mathcal{L}^{-1}\left[\frac{s}{(s^2+a^2)^2}\right] = \frac{t \sin at}{2a}$$

```
      SUBROUTINE VALUE (X,Y,T,AL,U,V,MMMMM)
C     THIS IS TABLE ENTRY NUMBER 19 OF LAPLACE TRANSFORMS BY SPIEGEL
      IMPLICIT REAL*8 (A,B,D-H,O-Z)
      IMPLICIT COMPLEX*16 (C)
      MMMMM = MMMMM + 1
      ZERO = 0.D+0
      CS1 = DCMPLX(X,Y)
      XPTR = X*T
      XPTI = Y*T
      CXPT = DCMPLX(XPTR,XPTI)
      CS3 = CDEXP(CXPT)
      CNUM = CS1*CS3
      CS4 = CS1*CS1
      CS5 = DCMPLX(AL,ZERO)
      CS6 = CS5*CS5
      CS7 = CS4 + CS6
      CDENOM = CS7*CS7
      CRESLT = CNUM/CDENOM
      U = DREAL(CRESLT)
      V = DIMAG(CRESLT)
      RETURN
      END
```

AL = 1.25D-1

T = 8.D+0

AA = 0.3

$X_F$ = -4.54

$y_F$ = 2.2

MMMMM = 23

LOSC = 3

SUM = 2.69271D+1

SUMA = 2.69278D+1

TIME = 0.89180D-1

ANALYTICAL    2.69271D+1

Case 20.

$$\mathcal{L}^{-1}\left[\frac{s^2}{(s^2+a^2)^2}\right] = \frac{\sin at + at \cos at}{2a}$$

```
      SUBROUTINE VALUE (X,Y,T,AL,U,V,MMMMM)
C     THIS IS TABLE ENTRY NUMBER 20 OF LAPLACE TRANSFORMS BY SPIEGEL
      IMPLICIT REAL*8 (A,B,D-H,O-Z)
      IMPLICIT COMPLEX*16 (C)
      MMMMM = MMMMM + 1
      ZERO = 0.D+0
      CS1 = DCMPLX(X,Y)
      CS2 = CS1*CS1
       XPTR = X*T
      XPTI = Y*T
      CXPT = DCMPLX(XPTR,XPTI)
      CS3 = CDEXP(CXPT)
      CNUM = CS2*CS3
      CS4 = CS1*CS1
      CS5 = DCMPLX(AL,ZERO)
      CS6 = CS5*CS5
      CS7 = CS4 + CS6
      CDENOM = CS7*CS7
      CRESLT = CNUM/CDENOM
      U = DREAL(CRESLT)
      V = DIMAG(CRESLT)
      RETURN
      END
```

AL = 1.25D-1

T = 8.D+0

AA = 0.3

$X_P$ = -4.54

$y_P$ = 2.2

MMMMM = 23

LOSC = 4

SUM = 5.52709D+0

SUMA = 5.89548D+0

TIME = 0.60684D-1

ANALYTICAL 5.52709D+0

Case 21.

$$\mathcal{L}^{-1}\left[\frac{s^3}{(s^2+a^2)^2}\right] = \cos at - \tfrac{1}{2}at \sin at$$

```
      SUBROUTINE VALUE (X,Y,T,AL,U,V,MMMMM)
C     THIS IS TABLE ENTRY NUMBER 21 OF LAPLACE TRANSFORMS BY SPIEGEL
      IMPLICIT REAL*8 (A,B,D-H,O-Z)
      IMPLICIT COMPLEX*16 (C)
      MMMMM = MMMMM + 1
      ZERO = 0.D+0
      CS1 = DCMPLX(X,Y)
      CS1 = DCMPLX(X,Y)
      CS2 = CS1*CS1*CS1
      XPTR = X*T
      XPTI = Y*T
      CXPT = DCMPLX(XPTR,XPTI)
      CS3 = CDEXP(CXPT)
      CNUM = CS2*CS3
      CS4 = CS1*CS1
      CS5 = DCMPLX(AL,ZERO)
      CS6 = CS5*CS5
      CS7 = CS4 + CS6
      CDENOM = CS7*CS7
      CRESLT = CNUM/CDENOM
      U = DREAL(CRESLT)
      V = DIMAG(CRESLT)
      RETURN
      END
```

AL = 1.25D-1

T = 8.D+0

AA = 0.3

$X_P$ = -4.99

$Y_P$ = 2.3

MMMMM = 24

LOSC = 5

SUM = 1.19567D-1

SUMA = 1.89481D+0

TIME = 0.10865D+0

ANALYTICAL 1.19567D-1

Case 22.

$$\mathcal{L}^{-1}\left[\frac{s^2 - a^2}{(s^2 + a^2)^2}\right] = t \cos at$$

```
      SUBROUTINE VALUE (X,Y,T,AL,U,V,MMMMM)
C     THIS IS TABLE ENTRY NUMBER 22 OF LAPLACE TRANSFORMS BY SPIEGEL
      IMPLICIT REAL*8 (A,B,D-H,O-Z)
      IMPLICIT COMPLEX*16 (C)
      MMMMM = MMMMM + 1
      ZERO = 0.D+0
      CS1 = DCMPLX(X,Y)
      CS2 = DCMPLX(AL,ZERO)
      XPTR = X*T
      XPTI = Y*T
      CXPT = DCMPLX(XPTR,XPTI)
      CNUM = ((CS1*CS1) - (CS2*CS2))*CDEXP(CXPT)
      CS3 = ((CS1*CS1) + (CS2*CS2))
      CDENOM = CS3*CS3
      CRESLT = CNUM/CDENOM
      U = DREAL(CRESLT)
      V = DIMAG(CRESLT)
      RETURN
      END
```

AL = 1.25D-1

T = 8.D+0

AA = 0.3

Xp = -4.54

yp = 2.2

MMMMM = 23

LOSC = 4

SUM = 4.32242D+0

SUMA = 4.76000D+0

TIME = 0.67210D-1

ANALYTICAL 4.32242D+0

Case 23.

$$\mathcal{L}^{-1}\left[\frac{1}{(s^2-a^2)^2}\right] = \frac{at\cosh at - \sinh at}{2a^3}$$

```
      SUBROUTINE VALUE (X,Y,T,AL,U,V,MMMMM)
C     THIS IS TABLE ENTRY NUMBER 23 OF LAPLACE TRANSFORMS BY SPIEGEL
      IMPLICIT REAL*8 (A,B,D-H,O-Z)
      IMPLICIT COMPLEX*16 (C)
      MMMMM = MMMMM + 1
      ZERO = 0.D+0
      CS1 = DCMPLX(X,Y)
      XPTR = X*T
      XPTI = Y*T
      CXPT = DCMPLX(XPTR,XPTI)
      CNUM = CDEXP(CXPT)
      CS4 = CS1*CS1
      CS5 = DCMPLX(AL,ZERO)
      CS6 = CS5*CS5
      CS7 = CS4 - CS6
      CDENOM = CS7*CS7
      CRESLT = CNUM/CDENOM
      U = DREAL(CRESLT)
      V = DIMAG(CRESLT)
      RETURN
      END
```

AL = 1.25D-1

T = 8.D+0

AA = 0.4

$X_F$ = -4.44

$y_F$ = 2.2

MMMMM = 23

LOSC = 2

SUM = 9.41771D+1

SUMA = 9.41771D+1

TIME = 0.61984D-1

ANALYTICAL 9.41771D+1

Case 24.

$$\mathcal{L}^{-1}\left[\frac{s}{(s^2-a^2)^2}\right] = \frac{t \sinh at}{2a}$$

```
      SUBROUTINE VALUE (X,Y,T,AL,U,V,MMMMM)
C     THIS IS TABLE ENTRY NUMBER 24 OF LAPLACE TRANSFORMS BY SPIEGEL
      IMPLICIT REAL*8 (A,B,D-H,O-Z)
      IMPLICIT COMPLEX*16 (C)
      MMMMM = MMMMM + 1
      ZERO = 0.D+0
      CS1 = DCMPLX(X,Y)
      XPTR = X*T
      XPTI = Y*T
      CXPT = DCMPLX(XPTR,XPTI)
      CS3 = CDEXP(CXPT)
      CNUM = CS1*CS3
      CS4 = CS1*CS1
      CS5 = DCMPLX(AL,ZERO)
      CS6 = CS5*CS5
      CS7 = CS4 - CS6
      CDENOM = CS7*CS7
      CRESLT = CNUM/CDENOM
      U = DREAL(CRESLT)
      V = DIMAG(CRESLT)
      RETURN
      END
```

AL = 1.25D-1

T = 8.D+0

AA = 0.4

$X_P$ = -4.44

$y_P$ = 2.2

MMMMM = 23

LOSC = 3

SUM = 3.76064D+1

SUMA = 3.76151D+1

TIME = 0.66924D-1

ANALYTICAL 3.76064D+1

Case 25.

$$\mathcal{L}^{-1}\left[\frac{s^2}{(s^2-a^2)^2}\right] = \frac{\sinh at + at \cosh at}{2a}$$

```
      SUBROUTINE VALUE (X,Y,T,AL,U,V,MMMMM)
C     THIS IS TABLE ENTRY NUMBER 25 OF LAPLACE TRANSFORMS BY SPIEGEL
      IMPLICIT REAL*8 (A,B,D-H,O-Z)
      IMPLICIT COMPLEX*16 (C)
      MMMMM = MMMMM + 1
      ZERO = 0.D+0
      CS1 = DCMPLX(X,Y)
      CS2 = CS1*CS1
       XPTR = X*T
      XPTI = Y*T
      CXPT = DCMPLX(XPTR,XPTI)
      CS3 = CDEXP(CXPT)
      CNUM = CS2*CS3
      CS4 = CS1*CS1
      CS5 = DCMPLX(AL,ZERO)
      CS6 = CS5*CS5
      CS7 = CS4 - CS6
      CDENOM = CS7*CS7
      CRESLT = CNUM/CDENOM
      U = DREAL(CRESLT)
      V = DIMAG(CRESLT)
      RETURN
      END
```

AL = 1.25D-1

T = 8.D+0

AA = 0.4

$X_P$ = -4.89

$y_P$ = 2.3

MMMMM = 24

LOSC = 4

SUM = 1.08731D+1

SUMA = 1.20870D+1

TIME = 0.73710D-1

ANALYTICAL = 1.08731D+1

Case 26.

$$\mathcal{L}^{-1}\left[\frac{s^3}{(s^2-a^2)^2}\right] = \cosh at + \tfrac{1}{2}at\sinh at$$

```
      SUBROUTINE VALUE (X,Y,T,AL,U,V,MMMMM)
C     THIS IS TABLE ENTRY NUMBER 26 OF LAPLACE TRANSFORMS BY SPIEGEL
      IMPLICIT REAL*8 (A,B,D-H,O-Z)
      IMPLICIT COMPLEX*16 (C)
      MMMMM = MMMMM + 1
      ZERO = 0.D+0
      CS1 = DCMPLX(X,Y)
      CS2 = CS1*CS1*CS1
      XPTR = X*T
      XPTI = Y*T
      CXPT = DCMPLX(XPTR,XPTI)
      CS3 = CDEXP(CXPT)
      CNUM = CS2*CS3
      CS4 = CS1*CS1
      CS5 = DCMPLX(AL,ZERO)
      CS6 = CS5*CS5
      CS7 = CS4 - CS6
      CDENOM = CS7*CS7
      CRESLT = CNUM/CDENOM
      U = DREAL(CRESLT)
      V = DIMAG(CRESLT)
      RETURN
      END
```

AL = 1.25D-1

T = 8.D+0

AA = 0.4

$X_P$ = -4.89

$y_P$ = 2.3

MMMMM = 24

LOSC = 5

SUM = 2.13068D+0

SUMA = 4.30908D+0

TIME = 0.19490D+0

ANALYTICAL 2.13068D+0

Case 27.

$$\mathcal{L}^{-1}\left[\frac{s^2+a^2}{(s^2-a^2)^2}\right] = t\cosh at$$

```
      SUBROUTINE VALUE (X,Y,T,AL,U,V,MMMMM)
C     THIS IS TABLE ENTRY NUMBER 27 OF LAPLACE TRANSFORMS BY SPIEGEL
      IMPLICIT REAL*8 (A,B,D-H,O-Z)
      IMPLICIT COMPLEX*16 (C)
      MMMMM = MMMMM + 1
      ZERO = 0.D+0
      CS1 = DCMPLX(X, Y)
      CS2 = DCMPLX(AL, ZERO)
      XPTR = X*T
      XPTI = Y*T
      CXPT = DCMPLX(XPTR,XPTI)
      CNUM = ((CS1*CS1) + (CS2*CS2))*CDEXP(CXPT)
      CS3 = ((CS1*CS1) - (CS2*CS2))
      CDENOM = CS3*CS3
      CRESLT = CNUM/CDENOM
      U = DREAL(CRESLT)
      V = DIMAG(CRESLT)
      RETURN
      END
```

AL = 1.25D-1

T = 8.D+0

AA = 0.4

$X_F$ = -4.89

$y_F$ = 2.3

MMMMM = 24

LOSC = 4

SUM = 1.23446D+1

SUMA = 1.34023D+1

TIME = 0.65078D-1

ANALYTICAL 1.23446D+1

Case 28.

$$\mathcal{L}^{-1}\left[\frac{1}{(s^2+a^2)^3}\right] = \frac{(3-a^2t^2)\sin at - 3at\cos at}{8a^5}$$

```
      SUBROUTINE VALUE (X,Y,T,AL,U,V,MMMMM)
C     THIS IS TABLE ENTRY NUMBER 28 OF LAPLACE TRANSFORMS BY SPIEGEL
      IMPLICIT REAL*8 (A,B,D-H,O-Z)
      IMPLICIT COMPLEX*16 (C)
      MMMMM = MMMMM + 1
      ZERO = 0.D+0
      CS1 = DCMPLX(X,Y)
      XPTR = X*T
      XPTI = Y*T
      CXPT = DCMPLX(XPTR,XPTI)
      CNUM = CDEXP(CXPT)
      CS4 = CS1*CS1
      CS5 = DCMPLX(AL,ZERO)
      CS6 = CS5*CS5
      CS7 = CS4 + CS6
      CDENOM = CS7*CS7*CS7
      CRESLT = CNUM/CDENOM
      U = DREAL(CRESLT)
      V = DIMAG(CRESLT)
      RETURN
      END
```

AL = 1.25D-1

T = 8.D+0

AA = 0.4

$x_F$ = -3.6

$y_F$ = 2.0

MMMMM = 21

LOSC = 2

SUM = 2.54096D+2

SUMA = 2.89148D+2

TIME = 0.50414D-1

ANALYTICAL = 2.54096D+2

Case 29.

$$\mathcal{L}^{-1}\left[\frac{s}{(s^2+a^2)^3}\right] = \frac{t \sin at - at^2 \cos at}{8a^3}$$

```
      SUBROUTINE VALUE (X,Y,T,AL,U,V,MMMMM)
C     THIS IS TABLE ENTRY NUMBER 29 OF LAPLACE TRANSFORMS BY SPIEGEL
      IMPLICIT REAL*8 (A,B,D-H,O-Z)
      IMPLICIT COMPLEX*16 (C)
      MMMMM = MMMMM + 1
      ZERO = 0.D+0
      CS1 = DCMPLX(X,Y)
      XPTR = X*T
      XPTI = Y*T
      CXPT = DCMPLX(XPTR,XPTI)
      CS3 = CDEXP(CXPT)
      CNUM = CS1*CS3
      CS4 = CS1*CS1
      CS5 = DCMPLX(AL,ZERO)
      CS6 = CS5*CS5
      CS7 = CS4 + CS6
      CDENOM = CS7*CS7*CS7
      CRESLT = CNUM/CDENOM
      U = DREAL(CRESLT)
      V = DIMAG(CRESLT)
      RETURN
      END
```

AL = 1.25D-1

T = 8.D+0

AA = 0.3

$x_F$ = -4.11

$y_F$ = 2.1

MMMMM = 22

LOSC = 3

SUM = 1.54198D+2

SUMA = 1.66758D+2

TIME = 0.56472D-1

ANALYTICAL = 1.54198D+2

Case 30.

$$\mathcal{L}^{-1}\left[\frac{s^2}{(s^2+a^2)^3}\right] = \frac{(1+a^2t^2)\sin at - at\cos at}{8a^3}$$

```
      SUBROUTINE VALUE (X,Y,T,AL,U,V,MMMMM)
C     THIS IS TABLE ENTRY NUMBER 30 OF LAPLACE TRANSFORMS BY SPIEGEL
      IMPLICIT REAL*8 (A,B,D-H,O-Z)
      IMPLICIT COMPLEX*16 (C)
      MMMMM = MMMMM + 1
      ZERO = 0.D+0
      CS1 = DCMPLX(X,Y)
      CS2 = CS1*CS1
      XPTR = X*T
      XPTI = Y*T
      CXPT = DCMPLX(XPTR,XPTI)
      CS3 = CDEXP(CXPT)
      CNUM = CS2*CS3
      CS4 = CS1*CS1
      CS5 = DCMPLX(AL,ZERO)
      CS6 = CS5*CS5
      CS7 = CS4 + CS6
      CDENOM = CS7*CS7*CS7
      CRESLT = CNUM/CDENOM
      U = DREAL(CRESLT)
      V = DIMAG(CRESLT)
      RETURN
      END
```

AL = 1.25D-1

T = 8.D+0

AA = 0.3

$X_F$ = -4.11

$y_F$ = 2.1

MMMMM = 22

LOSC = 2

SUM = 7.31289D+1

SUMA = 7.31289D+1

TIME = 0.57122D-1

ANALYTICAL = 7.31289D+1

Case 31.

$$\mathcal{L}^{-1}\left[\frac{s^3}{(s^2+a^2)^3}\right] = \frac{3t\sin at + at^2\cos at}{8a}$$

```
      SUBROUTINE VALUE (X,Y,T,AL,U,V,MMMMM)
C     THIS IS TABLE ENTRY NUMBER 31 OF LAPLACE TRANSFORMS BY SPIEGEL
      IMPLICIT REAL*8 (A,B,D-H,O-Z)
      IMPLICIT COMPLEX*16 (C)
      MMMMM = MMMMM + 1
      ZERO = 0.D+0
      CS1 = DCMPLX(X,Y)
      CS2 = CS1*CS1*CS1
      XPTR = X*T
      XPTI = Y*T
      CXPT = DCMPLX(XPTR,XPTI)
      CS3 = CDEXP(CXPT)
      CNUM = CS2*CS3
      CS4 = CS1*CS1
      CS5 = DCMPLX(AL,ZERO)
      CS6 = CS5*CS5
      CS7 = CS4 + CS6
      CDENOM = CS7*CS7*CS7
      CRESLT = CNUM/CDENOM
      U = DREAL(CRESLT)
      V = DIMAG(CRESLT)
      RETURN
      END
```

AL = 1.25D-1

T = 8.D+0

AA = 0.3

$X_F$ = -4.54

$y_F$ = 2.2

MMMMM = 23

LOSC = 3

SUM = 2.45177D+1

SUMA = 2.45184D+1

TIME = 0.60554D-1

ANALYTICAL = 2.45177D+1

Case 32.

$$\mathcal{L}^{-1}\left[\frac{s^4}{(s^2+a^2)^3}\right] = \frac{(3-a^2t^2)\sin at + 5at\cos at}{8a}$$

```
      SUBROUTINE VALUE (X,Y,T,AL,U,V,MMMMM)
C     THIS IS TABLE ENTRY NUMBER 32 OF LAPLACE TRANSFORMS BY SPIEGEL
      IMPLICIT REAL*8 (A,B,D-H,O-Z)
      IMPLICIT COMPLEX*16 (C)
      MMMMM = MMMMM + 1
      ZERO = 0.D+0
      CS1 = DCMPLX(X,Y)
      CS2 = CS1*CS1*CS1*CS1
      XPTR = X*T
      XPTI = Y*T
      CXPT = DCMPLX(XPTR,XPTI)
      CS3 = CDEXP(CXPT)
      CNUM = CS2*CS3
      CS4 = CS1*CS1
      CS5 = DCMPLX(AL,ZERO)
      CS6 = CS5*CS5
      CS7 = CS4 + CS6
      CDENOM = CS7*CS7*CS7
      CRESLT = CNUM/CDENOM
      U = DREAL(CRESLT)
      V = DIMAG(CRESLT)
      RETURN
      END
```

AL = 1.25D-1

T = 8.D+0

AA = 0.3

$X_P$ = -4.54

$y_P$ = 2.2

MMMMM = 23

LOSC = 4

SUM = 4.38445D+0

SUMA = 4.83048D+0

TIME = 6.7548D-2

ANALYTICAL = 4.38445D+0

Case 33.

$$\mathcal{L}^{-1}\left[\frac{s^5}{(s^2+a^2)^3}\right] = \frac{(8-a^2t^2)\cos at - 7at\sin at}{8}$$

```
      SUBROUTINE VALUE (X,Y,T,AL,U,V,MMMMM)
C     THIS IS TABLE ENTRY NUMBER 33 OF LAPLACE TRANSFORMS BY SPIEGEL
      IMPLICIT REAL*8 (A,B,D-H,O-Z)
      IMPLICIT COMPLEX*16 (C)
      MMMMM = MMMMM + 1
      ZERO = 0.D+0
      CS1 = DCMPLX(X,Y)
      CS2 = CS1*CS1*CS1*CS1*CS1
      XPTR = X*T
      XPTI = Y*T
      CXPT = DCMPLX(XPTR,XPTI)
      CS3 = CDEXP(CXPT)
      CNUM = CS2*CS3
      CS4 = CS1*CS1
      CS5 = DCMPLX(AL,ZERO)
      CS6 = CS5*CS5
      CS7 = CS4 + CS6
      CDENOM = CS7*CS7*CS7
      CRESLT = CNUM/CDENOM
      U = DREAL(CRESLT)
      V = DIMAG(CRESLT)
      RETURN
      END
```

AL = 1.25D-1

T = 8.D+0

AA = 0.3

$X_F$ = -4.99

$y_F$ = 2.3

MMMMM = 24

LOSC = 5

SUM = -2.63523D-1

SUMA = 1.81943D+0

TIME = 6.74414D-2

ANALYTICAL = -2.63523D-1

Case 34.

$$\mathcal{L}^{-1}\left[\frac{3s^2 - a^2}{(s^2 + a^2)^3}\right] = \frac{t^2 \sin at}{2a}$$

```
      SUBROUTINE VALUE (X,Y,T,AL,U,V,MMMMM)
C     THIS IS TABLE ENTRY NUMBER 34 OF LAPLACE TRANSFORMS BY SPIEGEL
      IMPLICIT REAL*8 (A,B,D-H,O-Z)
      IMPLICIT COMPLEX*16 (C)
      MMMMM = MMMMM + 1
      ZERO = 0.D+0
      CS1 = DCMPLX(X,Y)
      CS2 = DCMPLX(AL,ZERO)
      XPTR = X*T
      XPTI = Y*T
      CXPT = DCMPLX(XPTR,XPTI)
      CNUM = ((3.D+0*(CS1*CS1))-(CS2*CS2))*CDEXP(CXPT)
      CS3 = (CS1*CS1) + (CS2*CS2)
      CDENOM = CS3*CS3*CS3
      CRESLT = CNUM/CDENOM
      U = DREAL(CRESLT)
      V = DIMAG(CRESLT)
      RETURN
      END
```

AL = 1.25D-1

T = 8.D+0

AA = 0.4

$x_p$ = -4.44

$y_p$ = 2.2

MMMMM = 23

LOSC = 2

SUM = 2.15417D+2

SUMA = 2.15417D+2

TIME = 0.67834D-1

ANALYTICAL = 2.15147D+2

Case 35.

$$\mathcal{L}^{-1}\left[\frac{s^3 - 3a^2 s}{(s^2+a^2)^3}\right] = \frac{1}{2}t^2 \cos at$$

```
      SUBROUTINE VALUE (X,Y,T,AL,U,V,MMMMM)
C     THIS IS TABLE ENTRY NUMBER 35 OF LAPLACE TRANSFORMS BY SPIEGEL
      IMPLICIT REAL*8 (A,B,D-H,O-Z)
      IMPLICIT COMPLEX*16 (C)
      MMMMM = MMMMM + 1
      ZERO = 0.D+0
      CS1 = DCMPLX(X,Y)
      CS2 = DCMPLX(AL,ZERO)
      XPTR = X*T
      XPTI = Y*T
      CXPT = DCMPLX(XPTR,XPTI)
      CNUM = ((CS1*CS1*CS1)-(3.D+0*CS1*CS2*CS2))*CDEXP(CXPT)
      CS3 = (CS1*CS1) + (CS2*CS2)
      CDENOM = CS3*CS3*CS3
      CRESLT = CNUM/CDENOM
      U = DREAL(CRESLT)
      V = DIMAG(CRESLT)
      RETURN
      END
```

AL = 1.25D-1

T = 8.D+0

AA = 0.3

$X_P$ = -4.54

$y_P$ = 2.2

MMMMM = 23

LOSC = 3

SUM = 1.72897D+1

SUMA = 1.72904D+1

TIME = 8.4474D-2

ANALYTICAL = 1.72897D+1

Case 36.

$$\mathcal{L}^{-1}\left[\frac{s^4 - 6a^2s^2 + a^4}{(s^2 + a^2)^4}\right] = \frac{1}{6}t^3 \cos at$$

```
      SUBROUTINE VALUE (X,Y,T,AL,U,V,MMMMM)
C     THIS IS TABLE ENTRY NUMBER 36 OF LAPLACE TRANSFORMS BY SPIEGEL
      IMPLICIT REAL*8 (A,B,D-H,O-Z)
      IMPLICIT COMPLEX*16 (C)
      MMMMM = MMMMM + 1
      ZERO = 0.D+0
      CS1 = DCMPLX(X,Y)
      CS2 = DCMPLX(AL,ZERO)
      CS3 = CS1*CS1*CS1*CS1
      CS4 = 6.D+0*CS2*CS2*CS1*CS1
      CS5 = CS2*CS2*CS2*CS2
      XPTR = X*T
      XPTI = Y*T
      CXPT = DCMPLX(XPTR,XPTI)
      CNUM = (CS3-CS4+CS5)*CDEXP(CXPT)
      CS6 = ((CS1*CS1) + (CS2*CS2))
      CDENOM = CS6*CS6*CS6*CS6
      CRESLT = CNUM/CDENOM
      U = DREAL(CRESLT)
      V = DIMAG(CRESLT)
      RETURN
      END
```

AL = 1.25D-1

T = 8.D+0

AA =0.4

$X_P$ = -4.44

$y_P$ = 2.2

MMMMM = 23

LOSC = 2

SUM = 4.61058D+1

SUMA = 4.61058D+1

TIME = 0.73424D-1

ANALYTICAL = 4.61058D+1

Case 37.

$$\mathcal{L}^{-1}\left[\frac{s^3 - a^2 s}{(s^2 + a^2)^4}\right] = \frac{t^3 \sin at}{24a}$$

```
      SUBROUTINE VALUE (X,Y,T,AL,U,V,MMMMM)
C     THIS IS TABLE ENTRY NUMBER 37 OF LAPLACE TRANSFORMS BY SPIEGEL
      IMPLICIT REAL*8 (A,B,D-H,O-Z)
      IMPLICIT COMPLEX*16 (C)
      MMMMM = MMMMM + 1
      ZERO = 0.D+0
      CS1 = DCMPLX(X,Y)
      CS2 = DCMPLX(AL,ZERO)
      XPTR = X*T
      XPTI = Y*T
      CXPT = DCMPLX(XPTR,XPTI)
      CNUM = ((CS1*CS1*CS1) - (CS2*CS2*CS1))*CDEXP(CXPT)
      CS3 = ((CS1*CS1) + (CS2*CS2))
      CDENOM = CS3*CS3*CS3*CS3
      CRESLT = CNUM/CDENOM
      U = DREAL(CRESLT)
      V = DIMAG(CRESLT)
      RETURN
      END
```

AL = 1.25D-1

T = 8.D+0

AA = 0.4

$X_F$ = -4.01

$y_F$ = 2.1

MMMMM = 22

LOSC = 3

SUM = 1.43611D+2

SUMA =1.44005D+2

TIME = 0.68692D-1

ANALYTICAL = 1.43611D+2

Case 38.

$$\mathcal{L}^{-1}\left[\frac{1}{(s^2-a^2)^3}\right] = \frac{(3+a^2t^2)\sinh at - 3at\cosh at}{8a^5}$$

```
      SUBROUTINE VALUE (X,Y,T,AL,U,V,MMMMM)
C     THIS IS TABLE ENTRY NUMBER 38 OF LAPLACE TRANSFORMS BY SPIEGEL
      IMPLICIT REAL*8 (A,B,D-H,O-Z)
      IMPLICIT COMPLEX*16 (C)
      MMMMM = MMMMM + 1
      ZERO = 0.D+0
      CS1 = DCMPLX(X,Y)
      XPTR = X*T
      XPTI = Y*T
      CXPT = DCMPLX(XPTR,XPTI)
      CNUM = CDEXP(CXPT)
      CS4 = CS1*CS1
      CS5 = DCMPLX(AL,ZERO)
      CS6 = CS5*CS5
      CS7 = CS4 - CS6
      CDENOM = CS7*CS7*CS7
      CRESLT = CNUM/CDENOM
      U = DREAL(CRESLT)
      V = DIMAG(CRESLT)
      RETURN
      END
```

AL = 1.25D-1

T = 8.D+0

AA = 0.4

$X_P$ = -4.01

$y_P$ = 2.1

MMMMM = 22

LOSC = 2

SUM = 2.93122D+2

SUMA = 3.67471D+2

TIME = 0.48334D-1

ANALYTICAL = 2.93122D+2

Case 39.

$$\mathcal{L}^{-1}\left[\frac{s}{(s^2-a^2)^3}\right] = \frac{at^2\cosh at - t\sinh at}{8a^3}$$

```
      SUBROUTINE VALUE (X,Y,T,AL,U,V,MMMMM)
C     THIS IS TABLE ENTRY NUMBER 39 OF LAPLACE TRANSFORMS BY SPIEGEL
      IMPLICIT REAL*8 (A,B,D-H,O-Z)
      IMPLICIT COMPLEX*16 (C)
      MMMMM = MMMMM + 1
      ZERO = 0.D+0
      CS1 = DCMPLX(X,Y)
      XPTR = X*T
      XPTI = Y*T
      CXPT = DCMPLX(XPTR,XPTI)
      CS3 = CDEXP(CXPT)
      CNUM = CS1*CS3
      CS4 = CS1*CS1
      CS5 = DCMPLX(AL,ZERO)
      CS6 = CS5*CS5
      CS7 = CS4 - CS6
      CDENOM = CS7*CS7*CS7
      CRESLT = CNUM/CDENOM
      U = DREAL(CRESLT)
      V = DIMAG(CRESLT)
      RETURN
      END
```

AL = 1.25D-1

T = 8.D+0

AA = 0.4

$X_P$ = -4.01

$y_P$ = 2.1

MMMMM = 22

LOSC = 3

SUM = 1.88354D+2

SUMA = 1.89195D+2

TIME = 0.48750D-1

ANALYTICAL = 1.88354D+2

Case 40.

$$\mathcal{L}^{-1}\left[\frac{s^2}{(s^2-a^2)^3}\right] = \frac{at\cosh at + (a^2t^2-1)\sinh at}{8a^3}$$

```
      SUBROUTINE VALUE (X,Y,T,AL,U,V,MMMMM)
C     THIS IS TABLE ENTRY NUMBER 40 OF LAPLACE TRANSFORMS BY SPIEGEL
      IMPLICIT REAL*8 (A,B,D-H,O-Z)
      IMPLICIT COMPLEX*16 (C)
      MMMMM = MMMMM + 1
      ZERO = 0.D+0
      CS1 = DCMPLX(X,Y)
      CS2 = CS1*CS1
      XPTR = X*T
      XPTI = Y*T
      CXPT = DCMPLX(XPTR,XPTI)
      CS3 = CDEXP(CXPT)
      CNUM = CS2*CS3
      CS4 = CS1*CS1
      CS5 = DCMPLX(AL,ZERO)
      CS6 = CS5*CS5
      CS7 = CS4 - CS6
      CDENOM = CS7*CS7*CS7
      CRESLT = CNUM/CDENOM
      U = DREAL(CRESLT)
      V = DIMAG(CRESLT)
      RETURN
      END
```

AL = 1.25D-1

T = 8.D+0

AA = 0.4

$X_P$ = -4.44

$y_P$ = 2.2

MMMMM = 23

LOSC = 2

SUM = 9.87572D+1

SUMA = 9.87572D+1

TIME = 0.5252D-1

ANALYTICAL = 9.87572D+1

Case 41.

$$\mathcal{L}^{-1}\left[\frac{s^3}{(s^2-a^2)^3}\right] = \frac{3t\sinh at + at^2\cosh at}{8a}$$

```
C       THIS IS TABLE ENTRY NUMBER 41 OF LAPLACE TRANSFORMS BY SPIEGEL
        IMPLICIT REAL*8 (A,B,D-H,O-Z)
        IMPLICIT COMPLEX*16 (C)
        MMMMM = MMMMM + 1
        ZERO = 0.D+0
        CS1 = DCMPLX(X,Y)
        CS2 = CS1*CS1*CS1
        XPTR = X*T
        XPTI = Y*T
        CXPT = DCMPLX(XPTR,XPTI)
        CS3 = CDEXP(CXPT)
        CNUM = CS2*CS3
        CS4 = CS1*CS1
        CS5 = DCMPLX(AL,ZERO)
        CS6 = CS5*CS5
        CS7 = CS4 - CS6
        CDENOM = CS7*CS7*CS7
        CRESLT = CNUM/CDENOM
        U = DREAL(CRESLT)
        V = DIMAG(CRESLT)
        RETURN
        END
```

AL = 1.25D-1

T = 8.D+0

AA = 0.4

$X_F$ = -4.44

$y_F$ = 2.2

MMMMM = 23

LOSC = 3

SUM = 4.05495D+1

SUMA = 4.05582D+1

TIME = 0.53508D-1

ANALYTICAL = 4.05495D+1

Case 42.

$$\mathcal{L}^{-1}\left[\frac{s^4}{(s^2-a^2)^3}\right] = \frac{(3+a^2t^2)\sinh at + 5at\cosh at}{8a}$$

```
      SUBROUTINE VALUE (X,Y,T,AL,U,V,MMMMM)
C     THIS IS TABLE ENTRY NUMBER 42 OF LAPLACE TRANSFORMS BY SPIEGEL
      IMPLICIT REAL*8 (A,B,D-H,O-Z)
      IMPLICIT COMPLEX*16 (C)
      MMMMM = MMMMM + 1
      ZERO = 0.D+0
      CS1 = DCMPLX(X,Y)
      CS2 = CS1*CS1*CS1*CS1
      XPTR = X*T
      XPTI = Y*T
      CXPT = DCMPLX(XPTR,XPTI)
      CS3 = CDEXP(CXPT)
      CNUM = CS2*CS3
      CS4 = CS1*CS1
      CS5 = DCMPLX(AL,ZERO)
      CS6 = CS5*CS5
      CS7 = CS4 - CS6
      CDENOM = CS7*CS7*CS7
      CRESLT = CNUM/CDENOM
      U = DREAL(CRESLT)
      V = DIMAG(CRESLT)
      RETURN
      END
```

AL = 1.25D-1

T = 8.D+0

AA = 0.4

$X_P$ = -4.89

$y_P$ = 2.3

MMMMM = 24

LOSC = 4

SUM = 1.24162D+1

SUMA = 1.34824D+1

TIME = 0.58240D-1

ANALYTICAL = 1.24162D+1

Case 43.

$$\mathcal{L}^{-1}\left[\frac{s^5}{(s^2-a^2)^3}\right] = \frac{(8+a^2t^2)\cosh at + 7at\sinh at}{8}$$

```
      SUBROUTINE VALUE (X,Y,T,AL,U,V,MMMMM)
C     THIS IS TABLE ENTRY NUMBER 43 OF LAPLACE TRANSFORMS BY SPIEGEL
      IMPLICIT REAL*8 (A,B,D-H,O-Z)
      IMPLICIT COMPLEX*16 (C)
      MMMMM = MMMMM + 1
      ZERO = 0.D+0
      CS1 = DCMPLX(X,Y)
      CS2 = CS1*CS1*CS1*CS1*CS1
      XPTR = X*T
      XPTI = Y*T
      CXPT = DCMPLX(XPTR,XPTI)
      CS3 = CDEXP(CXPT)
      CNUM = CS2*CS3
      CS4 = CS1*CS1
      CS5 = DCMPLX(AL,ZERO)
      CS6 = CS5*CS5
      CS7 = CS4 - CS6
      CDENOM = CS7*CS7*CS7
      CRESLT = CNUM/CDENOM
      U = DREAL(CRESLT)
      V = DIMAG(CRESLT)
      RETURN
      END
```

AL = 1.25D-1

T = 8.D+0

AA = 0.4

$X_F$ = -4.89

$y_F$ = 2.3

MMMMM = 24

LOSC = 5

SUM = 2.76427D+0

SUMA = 4.73261D+0

TIME = 0.62920D-1

ANALYTICAL = 2.76427D+0

Case 44.

$$\mathcal{L}^{-1}\left[\frac{3s^2 + a^2}{(s^2 - a^2)^3}\right] = \frac{t^2 \sinh at}{2a}$$

```
      SUBROUTINE VALUE (X,Y,T,AL,U,V,MMMMM)
C     THIS IS TABLE ENTRY NUMBER 44 OF LAPLACE TRANSFORMS BY SPIEGEL
      IMPLICIT REAL*8 (A,B,D-H,O-Z)
      IMPLICIT COMPLEX*16 (C)
      MMMMM = MMMMM + 1
      ZERO = 0.D+0
      CS1 = DCMPLX(X,Y)
      CS2 = DCMPLX(AL,ZERO)
      XPTR = X*T
      XPTI = Y*T
      CXPT = DCMPLX(XPTR,XPTI)
      CNUM = ((3.D+0*(CS1*CS1))+(CS2*CS2))*CDEXP(CXPT)
      CS3 = (CS1*CS1) - (CS2*CS2)
      CDENOM = CS3*CS3*CS3
      CRESLT = CNUM/CDENOM
      U = DREAL(CRESLT)
      V = DIMAG(CRESLT)
      RETURN
      END
```

AL = 1.25D-1

T = 8.D+0

AA = 0.4

$X_F$ = -4.44

$y_F$ = 2.2

MMMMM = 23

LOSC = 2

SUM = 3.00852D+2

SUMA = 3.00852D+2

TIME = 0.55042D-1

ANALYTICAL = 3.00852D+2

Case 45.

$$\mathcal{L}^{-1}\left[\frac{s^3+3a^2s}{(s^2-a^2)^3}\right] = \tfrac{1}{2}t^2\cosh at$$

```
      SUBROUTINE VALUE (X,Y,T,AL,U,V,MMMMM)
C     THIS IS TABLE ENTRY NUMBER 45 OF LAPLACE TRANSFORMS BY SPIEGEL
      IMPLICIT REAL*8 (A,B,D-H,O-Z)
      IMPLICIT COMPLEX*16 (C)
      MMMMM = MMMMM + 1
      ZERO = 0.D+0
      CS1 = DCMPLX(X,Y)
      CS2 = DCMPLX(AL,ZERO)
      XPTR = X*T
      XPTI = Y*T
      CXPT = DCMPLX(XPTR,XPTI)
      CNUM = ((CS1*CS1*CS1)+(3.D+0*CS1*CS2*CS2))*CDEXP(CXPT)
      CS3 = (CS1*CS1) - (CS2*CS2)
      CDENOM = CS3*CS3*CS3
      CRESLT = CNUM/CDENOM
      U = DREAL(CRESLT)
      V = DIMAG(CRESLT)
      RETURN
      END
```

AL = 1.25D-1

T = 8.D+0

AA = 0.4

$X_F$ = -4.44

$y_F$ = 2.2

MMMMM = 23

LOSC = 3

SUM = 4.93786D+1

SUMA = 4.93892D+1

TIME = 0.57772D-1

ANALYTICAL = 4.93786D+1

Case 46.

$$\mathcal{L}^{-1}\left[\frac{s^4 + 6a^2s^2 + a^4}{(s^2 - a^2)^4}\right] = \tfrac{1}{6}t^3 \cosh at$$

```
      SUBROUTINE VALUE (X,Y,T,AL,U,V,MMMMM)
C     THIS IS TABLE ENTRY NUMBER 46 OF LAPLACE TRANSFORMS BY SPIEGEL
      IMPLICIT REAL*8 (A,B,D-H,O-Z)
      IMPLICIT COMPLEX*16 (C)
      MMMMM = MMMMM + 1
      ZERO = 0.D+0
      CS1 = DCMPLX(X,Y)
      CS2 = DCMPLX(AL,ZERO)
      CS3 = CS1*CS1*CS1*CS1
      CS4 = 6.D+0*CS2*CS2*CS1*CS1
      CS5 = CS2*CS2*CS2*CS2
      XPTR = X*T
      XPTI = Y*T
      CXPT = DCMPLX(XPTR,XPTI)
      CNUM = (CS3+CS4+CS5)*CDEXP(CXPT)
      CS6 = ((CS1*CS1) - (CS2*CS2))
      CDENOM = CS6*CS6*CS6*CS6
      CRESLT = CNUM/CDENOM
      U = DREAL(CRESLT)
      V = DIMAG(CRESLT)
      RETURN
      END
```

AL = 1.25D-1

T = 8.D+0

AA = 0.4

$X_P$ = -4.44

$y_P$ = 2.2

MMMMM = 23

LOSC = 2

SUM = 1.31676D+2

SUMA = 1.31676D+2

TIME = 0.65676D-1

ANALYTICAL = 1.31676D+2

Case 47.

$$\mathcal{L}^{-1}\left[\frac{s^3 + a^2 s}{(s^2 - a^2)^4}\right] = \frac{t^3 \sinh at}{24a}$$

```
      SUBROUTINE VALUE (X,Y,T,AL,U,V,MMMMM)
C     THIS IS TABLE ENTRY NUMBER 47 OF LAPLACE TRANSFORMS BY SPIEGEL
      IMPLICIT REAL*8 (A,B,D-H,O-Z)
      IMPLICIT COMPLEX*16 (C)
      MMMMM = MMMMM + 1
      ZERO = 0.D+0
      CS1 = DCMPLX(X,Y)
      CS2 = DCMPLX(AL,ZERO)
      XPTR = X*T
      XPTI = Y*T
      CXPT = DCMPLX(XPTR,XPTI)
      CNUM = ((CS1*CS1*CS1) + (CS2*CS2*CS1))*CDEXP(CXPT)
      CS3 = ((CS1*CS1) - (CS2*CS2))
      CDENOM = CS3*CS3*CS3*CS3
      CRESLT = CNUM/CDENOM
      U = DREAL(CRESLT)
      V = DIMAG(CRESLT)
      RETURN
      END
```

AL = 1.25D-1

T = 8.D+0

AA = 0.4

$X_P$ = -4.01

$y_P$ = 2.1

MMMMM = 22

LOSC = 3

SUM = 2.00568D+2

SUMA = 2.01796D+2

TIME = 0.55822D-1

ANALYTICAL = 2.00568D+2

Case 48.

$$\mathcal{L}^{-1}\left[\frac{1}{s^3+a^3}\right] = \frac{e^{at/2}}{3a^2}\left\{\sqrt{3}\sin\frac{\sqrt{3}at}{2} - \cos\frac{\sqrt{3}at}{2} + e^{-3at/2}\right\}$$

```
      SUBROUTINE VALUE (X,Y,T,AL,U,V,MMMMM)
C     THIS IS TABLE ENTRY NUMBER 48 OF LAPLACE TRANSFORMS BY SPIEGEL
      IMPLICIT REAL*8 (A,B,D-H,O-Z)
      IMPLICIT COMPLEX*16 (C)
      MMMMM = MMMMM + 1
      ZERO = 0.D+0
      CS1 = DCMPLX(X,Y)
      XPTR = X*T
      XPTI = Y*T
      CXPT = DCMPLX(XPTR,XPTI)
      CNUM = CDEXP(CXPT)
      CS4 = CS1*CS1*CS1
      CS5 = DCMPLX(AL,ZERO)
      CS6 = CS5*CS5*CS5
      CDENOM = CS4 + CS6
      CRESLT = CNUM/CDENOM
      U = DREAL(CRESLT)
      V = DIMAG(CRESLT)
      RETURN
      END
```

AL = 1.25D-1

T = 8.D+0

AA = 0.4

$X_P$ = -4.44

$y_F$ = 2.2

MMMMM = 23

LOSC = 3

SUM = 3.14683D+1

SUMA = 3.14767D+1

TIME = 0.48958D-1

ANALYTICAL = 3.14683D+2

Case 49.

$$\mathcal{L}^{-1}\left[\frac{s}{s^3+a^3}\right] = \frac{e^{at/2}}{3a}\left\{\cos\frac{\sqrt{3}\,at}{2} + \sqrt{3}\sin\frac{\sqrt{3}\,at}{2} - e^{-3at/2}\right\}$$

```
      SUBROUTINE VALUE (X,Y,T,AL,U,V,MMMMM)
C     THIS IS TABLE ENTRY NUMBER 49 OF LAPLACE TRANSFORMS BY SPIEGEL
      IMPLICIT REAL*8 (A,B,D-H,O-Z)
      IMPLICIT COMPLEX*16 (C)
      MMMMM = MMMMM + 1
      ZERO = 0.D+0
      CS1 = DCMPLX(X,Y)
      XPTR = X*T
      XPTI = Y*T
      CXPT = DCMPLX(XPTR,XPTI)
      CS3 = CDEXP(CXPT)
      CNUM = CS1*CS3
      CS4 = CS1*CS1*CS1
      CS5 = DCMPLX(AL,ZERO)
      CS6 = CS5*CS5*CS5
      CDENOM = CS4 + CS6
      CRESLT = CNUM/CDENOM
      U = DREAL(CRESLT)
      V = DIMAG(CRESLT)
      RETURN
      END
```

AL = 1.25D-1

T = 8.D+0

AA = 0.3

$X_P$ = -4.54

$y_P$ = 2.2

MMMMM = 23

LOSC = 4

SUM = 7.66825D+0

SUMA = 7.92326D+0

TIME = 0.52416D-1

ANALYTICAL = 7.66825D+0

Case 50.

$$\mathcal{L}^{-1}\left[\frac{s^2}{s^3+a^3}\right] = \frac{1}{3}\left(e^{-at} + 2e^{at/2}\cos\frac{\sqrt{3}\,at}{2}\right)$$

```
      SUBROUTINE VALUE (X,Y,T,AL,U,V,MMMMM)
C     THIS IS TABLE ENTRY NUMBER 50 OF LAPLACE TRANSFORMS BY SPIEGEL
      IMPLICIT REAL*8 (A,B,D-H,O-Z)
      IMPLICIT COMPLEX*16 (C)
      MMMMM = MMMMM + 1
      ZERO = 0.D+0
      CS1 = DCMPLX(X,Y)
      CS2 = CS1*CS1
      XPTR = X*T
      XPTI = Y*T
      CXPT = DCMPLX(XPTR,XPTI)
      CS3 = CDEXP(CXPT)
      CNUM = CS2*CS3
      CS4 = CS1*CS1*CS1
      CS5 = DCMPLX(AL,ZERO)
      CS6 = CS5*CS5*CS5
      CDENOM = CS4 + CS6
      CRESLT = CNUM/CDENOM
      U = DREAL(CRESLT)
      V = DIMAG(CRESLT)
      RETURN
      END
```

AL = 1.25D-1

T = 8.D+0

AA = 0.4

$x_F$ = -4.89

$y_F$ = 2.3

MMMMM = 24

LOSC = 5

SUM = 8.34720D-1

SUMA = 3.88818D+0

TIME = 0.55796D-1

ANALYTICAL = 8.34720D-1

Case 51.

$$\mathcal{L}^{-1}\left[\frac{1}{s^3 - a^3}\right] = \frac{e^{-at/2}}{3a^2}\left\{e^{3at/2} - \cos\frac{\sqrt{3}\,at}{2} - \sqrt{3}\sin\frac{\sqrt{3}\,at}{2}\right\}$$

```
      SUBROUTINE VALUE (X,Y,T,AL,U,V,MMMMM)
C     THIS IS TABLE ENTRY NUMBER 51 OF LAPLACE TRANSFORMS BY SPIEGEL
      IMPLICIT REAL*8 (A,B,D-H,O-Z)
      IMPLICIT COMPLEX*16 (C)
      MMMMM = MMMMM + 1
      ZERO = 0.D+0
      CS1 = DCMPLX(X,Y)
      XPTR = X*T
      XPTI = Y*T
      CXPT = DCMPLX(XPTR,XPTI)
      CNUM = CDEXP(CXPT)
      CS4 = CS1*CS1*CS1
      CS5 = DCMPLX(AL,ZERO)
      CS6 = CS5*CS5*CS5
      CDENOM = CS4 - CS6
      CRESLT = CNUM/CDENOM
      U = DREAL(CRESLT)
      V = DIMAG(CRESLT)
      RETURN
      END
```

AL = 1.25D-1

T = 8.D+0

AA = 0.4

$X_P$ = -4.44

$y_P$ = 2.2

MMMMM = 23

LOSC = 3

SUM = 3.25349D+1

SUMA = 3.25434D+1

TIME = 0.61204D-1

ANALYTICAL = 3.25349D+1

Case 52.

$$\mathcal{L}^{-1}\left[\frac{s}{s^3 - a^3}\right] = \frac{e^{-at/2}}{3a}\left\{\sqrt{3}\sin\frac{\sqrt{3}\,at}{2} - \cos\frac{\sqrt{3}\,at}{2} + e^{3at/2}\right\}$$

```
      SUBROUTINE VALUE (X,Y,T,AL,U,V,MMMMM)
C     THIS IS TABLE ENTRY NUMBER 52 OF LAPLACE TRANSFORMS BY SPIEGEL
      IMPLICIT REAL*8 (A,B,D-H,O-Z)
      IMPLICIT COMPLEX*16 (C)
      MMMMM = MMMMM + 1
      ZERO = 0.D+0
      CS1 = DCMPLX(X,Y)
      XPTR = X*T
      XPTI = Y*T
      CXPT = DCMPLX(XPTR,XPTI)
      CS3 = CDEXP(CXPT)
      CNUM = CS1*CS3
      CS4 = CS1*CS1*CS1
      CS5 = DCMPLX(AL,ZERO)
      CS6 = CS5*CS5*CS5
      CDENOM = CS4 - CS6
      CRESLT = CNUM/CDENOM
      U = DREAL(CRESLT)
      V = DIMAG(CRESLT)
      RETURN
      END
```

AL = 1.25D-1

T = 8.D+0

AA = 0.4

$x_F$ = -4.89

$y_F$ = 2.3

MMMMM = 24

LOSC = 4

SUM = 8.33492D+0

SUMA = 9.86283D+0

TIME = 0.67418D-1

ANALYTICAL = 8.33492D+0

Case 53.

$$\mathcal{L}^{-1}\left[\frac{s^2}{s^3 - a^3}\right] = \frac{1}{3}\left(e^{at} + 2e^{-at/2}\cos\frac{\sqrt{3}\,at}{2}\right)$$

```
      SUBROUTINE VALUE (X,Y,T,AL,U,V,MMMMM)
C     THIS IS TABLE ENTRY NUMBER 53 OF LAPLACE TRANSFORMS BY SPIEGEL
      IMPLICIT REAL*8 (A,B,D-H,O-Z)
      IMPLICIT COMPLEX*16 (C)
      MMMMM = MMMMM + 1
      ZERO = 0.D+0
      CS1 = DCMPLX(X,Y)
      CS2 = CS1*CS1
      XPTR = X*T
      XPTI = Y*T
      CXPT = DCMPLX(XPTR,XPTI)
      CS3 = CDEXP(CXPT)
      CNUM = CS2*CS3
      CS4 = CS1*CS1*CS1
      CS5 = DCMPLX(AL,ZERO)
      CS6 = CS5*CS5*CS5
      CDENOM = CS4 - CS6
      CRESLT = CNUM/CDENOM
      U = DREAL(CRESLT)
      V = DIMAG(CRESLT)
      RETURN
      END
```

AL = 1.25D-1

T = 8.D+0

AA = 0.3

$X_F$ = -4.99

$y_F$ = 2.3

MMMMM = 24

LOSC = 5

SUM = 1.16806D+0

SUMA = 2.25373D+0

TIME = 0.70200D-1

ANALYTICAL = 1.16806D+0

Case 54.

$$\mathcal{L}^{-1}\left[\frac{1}{s^4 + 4a^4}\right] = \frac{1}{4a^3}(\sin at \cosh at - \cos at \sinh at)$$

```
      SUBROUTINE VALUE (X,Y,T,AL,U,V,MMMMM)
C     THIS IS TABLE ENTRY NUMBER 54 OF LAPLACE TRANSFORMS BY SPIEGEL
      IMPLICIT REAL*8 (A,B,D-H,O-Z)
      IMPLICIT COMPLEX*16 (C)
      MMMMM = MMMMM + 1
      ZERO = 0.D+0
      CS1 = DCMPLX(X,Y)
      XPTR = X*T
      XPTI = Y*T
      CXPT = DCMPLX(XPTR,XPTI)
      CNUM = CDEXP(CXPT)
      CS4 = CS1*CS1*CS1*CS1
      CS5 = DCMPLX(AL,ZERO)
      CS6 = 4.D+0*(CS5*CS5*CS5*CS5)
      CDENOM = CS4 + CS6
      CRESLT = CNUM/CDENOM
      U = DREAL(CRESLT)
      V = DIMAG(CRESLT)
      RETURN
      END
```

AL = 1.25D-1

T = 8.D+0

AA = 0.4

$X_F$ = -4.44

$y_F$ = 2.2

MMMMM = 23

LOSC = 2

SUM = 8.49272D+1

SUMA = 8.49272D+1

TIME = 0.67782D-1

ANALYTICAL = 8.49272D+1

Case 55.

$$\mathcal{L}^{-1}\left[\frac{s}{s^4 + 4a^4}\right] = \frac{\sin at \sinh at}{2a^2}$$

```
      SUBROUTINE VALUE (X,Y,T,AL,U,V,MMMMM)
C     THIS IS TABLE ENTRY NUMBER 55 OF LAPLACE TRANSFORMS BY SPIEGEL
      IMPLICIT REAL*8 (A,B,D-H,O-Z)
      IMPLICIT COMPLEX*16 (C)
      MMMMM = MMMMM + 1
      ZERO = 0.D+0
      CS1 = DCMPLX(X,Y)
      XPTR = X*T
      XPTI = Y*T
      CXPT = DCMPLX(XPTR,XPTI)
      CS3 = CDEXP(CXPT)
      CNUM = CS1*CS3
      CS4 = CS1*CS1*CS1*CS1
      CS5 = DCMPLX(AL,ZERO)
      CS6 = 4.D+0*(CS5*CS5*CS5*CS5)
      CDENOM = CS4 + CS6
      CRESLT = CNUM/CDENOM
      U = DREAL(CRESLT)
      V = DIMAG(CRESLT)
      RETURN
      END
```

AL = 1.25D-1

T = 8.D+0

AA = 0.4

$x_F$ = -4.54

$y_F$ = 2.2

MMMMM = 23

LOSC = 3

SUM = 3.16447D+1

SUMA = 3.16532D+1

TIME = 0.59358D-1

ANALYTICAL = 3.16447D+1

Case 56.

$$\mathcal{L}^{-1}\left[\frac{s^2}{s^4+4a^4}\right] = \frac{1}{2a}(\sin at \cosh at + \cos at \sinh at)$$

```
      SUBROUTINE VALUE (X,Y,T,AL,U,V,MMMMM)
C     THIS IS TABLE ENTRY NUMBER 56 OF LAPLACE TRANSFORMS BY SPIEGEL
      IMPLICIT REAL*8 (A,B,D-H,O-Z)
      IMPLICIT COMPLEX*16 (C)
      MMMMM = MMMMM + 1
      ZERO = 0.D+0
      CS1 = DCMPLX(X,Y)
      CS2 = CS1*CS1
      XPTR = X*T
      XPTI = Y*T
      CXPT = DCMPLX(XPTR,XPTI)
      CS3 = CDEXP(CXPT)
      CNUM = CS2*CS3
      CS4 = CS1*CS1*CS1*CS1
      CS5 = DCMPLX(AL,ZERO)
      CS6 = 4.D+0*(CS5*CS5*CS5*CS5)
      CDENOM = CS4 + CS6
      CRESLT = CNUM/CDENOM
      U = DREAL(CRESLT)
      V = DIMAG(CRESLT)
      RETURN
      END
```

AL = 1.25D-1

T = 8.D+0

AA = 0.4

$X_F$ = -4.89

$y_F$ = 2.3

MMMMM = 24

LOSC = 4

SUM = 7.73369D+0

SUMA = 9.23364D+0

TIME = 0.76206D-1

ANALYTICAL = 7.73369D+0

Case 57.

$$\mathcal{L}^{-1}\left[\frac{s^3}{s^4+4a^4}\right] = \cos at \cosh at$$

```
      SUBROUTINE VALUE (X,Y,T,AL,U,V,MMMMM)
C     THIS IS TABLE ENTRY NUMBER 57 OF LAPLACE TRANSFORMS BY SPIEGEL
      IMPLICIT REAL*8 (A,B,D-H,O-Z)
      IMPLICIT COMPLEX*16 (C)
      MMMMM = MMMMM + 1
      ZERO = 0.D+0
      CS1 = DCMPLX(X,Y)
      CS2 = CS1*CS1*CS1
      XPTR = X*T
      XPTI = Y*T
      CXPT = DCMPLX(XPTR,XPTI)
      CS3 = CDEXP(CXPT)
      CNUM = CS2*CS3
      CS4 = CS1*CS1*CS1*CS1
      CS5 = DCMPLX(AL,ZERO)
      CS6 = 4.D+0*(CS5*CS5*CS5*CS5)
      CDENOM = CS4 + CS6
      CRESLT = CNUM/CDENOM
      U = DREAL(CRESLT)
      V = DIMAG(CRESLT)
      RETURN
      END
```

AL = 1.25D-1

T = 8.D+0

AA = 0.4

$X_F$ = -4.89

$y_F$ = 2.3

MMMMM = 24

LOSC = 5

SUM = 8.33730D-1

SUMA = 3.85232D+0

TIME = 0.82576D-1

ANALYTICAL = 8.33730D-1

Case 58.

$$\mathcal{L}^{-1}\left[\frac{1}{s^4 - a^4}\right] = \frac{1}{2a^3}(\sinh at - \sin at)$$

```
      SUBROUTINE VALUE (X,Y,T,AL,U,V,MMMMM)
C     THIS IS TABLE ENTRY NUMBER 58 OF LAPLACE TRANSFORMS BY SPIEGEL
      IMPLICIT REAL*8 (A,B,D-H,O-Z)
      IMPLICIT COMPLEX*16 (C)
      MMMMM = MMMMM + 1
      ZERO = 0.D+0
      CS1 = DCMPLX(X,Y)
      XPTR = X*T
      XPTI = Y*T
      CXPT = DCMPLX(XPTR,XPTI)
      CNUM = CDEXP(CXPT)
      CS4 = CS1*CS1*CS1*CS1
      CS5 = DCMPLX(AL,ZERO)
      CS6 = CS5*CS5*CS5*CS5
      CDENOM = CS4 - CS6
      CRESLT = CNUM/CDENOM
      U = DREAL(CRESLT)
      V = DIMAG(CRESLT)
      RETURN
      END
```

AL = 1.25D-1

T = 8.D+0

AA = 0.4

$X_P$ = -4.44

$y_P$ = 2.2

MMMMM = 23

LOSC = 2

SUM = 8.54349D+1

SUMA = 8.543490D+1

TIME = 0.74490D-1

ANALYTICAL = 8.54349D+1

Case 59.

$$\mathcal{L}^{-1}\left[\frac{s}{s^4 - a^4}\right] = \frac{1}{2a^2}(\cosh at - \cos at)$$

```
      SUBROUTINE VALUE (X,Y,T,AL,U,V,MMMMM)
C     THIS IS TABLE ENTRY NUMBER 59 OF LAPLACE TRANSFORMS BY SPIEGEL
      IMPLICIT REAL*8 (A,B,D-H,O-Z)
      IMPLICIT COMPLEX*16 (C)
      MMMMM = MMMMM + 1
      ZERO = 0.D+0
      CS1 = DCMPLX(X,Y)
      XPTR = X*T
      XPTI = Y*T
      CXPT = DCMPLX(XPTR,XPTI)
      CS3 = CDEXP(CXPT)
      CNUM = CS1*CS3
      CS4 = CS1*CS1*CS1*CS1
      CS5 = DCMPLX(AL,ZERO)
      CS6 = CS5*CS5*CS5*CS5
      CDENOM = CS4 - CS6
      CRESLT = CNUM/CDENOM
      U = DREAL(CRESLT)
      V = DIMAG(CRESLT)
      RETURN
      END
```

AL = 1.25D-1

T = 8.D+0

AA = 0.3

$x_F$ = -4.54

$y_F$ = 2.2

MMMMM = 23

LOSC = 3

SUM = 3.20889D+1

SUMA = 3.20897D+1

TIME = 0.78390D-1

ANALYTICAL = 3.20889D+1

Case 60.

$$\mathcal{L}^{-1}\left[\frac{s^2}{s^4 - a^4}\right] = \frac{1}{2a}(\sinh at + \sin at)$$

```
      SUBROUTINE VALUE (X,Y,T,AL,U,V,MMMMM)
C     THIS IS TABLE ENTRY NUMBER 60 OF LAPLACE TRANSFORMS BY SPIEGEL
      IMPLICIT REAL*8 (A,B,D-H,O-Z)
      IMPLICIT COMPLEX*16 (C)
      MMMMM = MMMMM + 1
      ZERO = 0.D+0
      CS1 = DCMPLX(X,Y)
      CS2 = CS1*CS1
      XPTR = X*T
      XPTI = Y*T
      CXPT = DCMPLX(XPTR,XPTI)
      CS3 = CDEXP(CXPT)
      CNUM = CS2*CS3
      CS4 = CS1*CS1*CS1*CS1
      CS5 = DCMPLX(AL,ZERO)
      CS6 = CS5*CS5*CS5*CS5
      CDENOM = CS4 - CS6
      CRESLT = CNUM/CDENOM
      U = DREAL(CRESLT)
      V = DIMAG(CRESLT)
      RETURN
      END
```

AL = 1.25D-1

T = 8.D+0

AA = 0.3

X = -4.54

y = 2.2

MMMMM = 23

LOSC = 4

SUM = 8.06669D+0

SUMA = 8.32767D+0

TIME = 0.87672D-1

ANALYTICAL = 8.06669D+0

Case 61.

$$\mathcal{L}^{-1}\left[\frac{s^3}{s^4 - a^4}\right] = \tfrac{1}{2}(\cosh at + \cos at)$$

```
      SUBROUTINE VALUE (X,Y,T,AL,U,V,MMMMM)
C     THIS IS TABLE ENTRY NUMBER 61 OF LAPLACE TRANSFORMS BY SPIEGEL
      IMPLICIT REAL*8 (A,B,D-H,O-Z)
      IMPLICIT COMPLEX*16 (C)
      MMMMM = MMMMM + 1
      ZERO = 0.D+0
      CS1 = DCMPLX(X,Y)
      CS2 = CS1*CS1*CS1
      XPTR = X*T
      XPTI = Y*T
      CXPT = DCMPLX(XPTR,XPTI)
      CS3 = CDEXP(CXPT)
      CNUM = CS2*CS3
      CS4 = CS1*CS1*CS1*CS1
      CS5 = DCMPLX(AL,ZERO)
      CS6 = CS5*CS5*CS5*CS5
      CDENOM = CS4 - CS6
      CRESLT = CNUM/CDENOM
      U = DREAL(CRESLT)
      V = DIMAG(CRESLT)
      RETURN
      END
```

AL = 1.25D-1

T = 8.D+0

AA = 0.3

$X_P$ = -4.99

$y_F$ = 2.3

MMMMM = 24

LOSC = 5

SUM = 1.04169D+0

SUMA = 2.18326D+0

TIME = 0.93314D-1

ANALYTICAL = 1.04169D+0

Case 62.

$$\mathcal{L}^{-1}\left[\frac{1}{\sqrt{s+a}+\sqrt{s+b}}\right] = \frac{e^{-bt}-e^{-at}}{2(b-a)\sqrt{\pi t^3}}$$

```
      SUBROUTINE VALUE (X, Y, T, AL, BE, U, V, MMMMM, NCALL)
C     THIS IS TABLE ENTRY NUMBER 62 OF LAPLACE TRANSFORMS BY SPIEGEL
      IMPLICIT REAL*8 (A,B,D-H,O-Z)
      IMPLICIT COMPLEX*16 (C)
      DIMENSION NCALL(5)
      MMMMM = MMMMM + 1
      ZERO = 0.D+0
      CS1 = DCMPLX(X,Y)
      CS2 = DCMPLX(AL,ZERO)
      CS3 = DCMPLX(BE,ZERO)
      XPTR = X*T
      XPTI = Y*T
      CXPT = DCMPLX(XPTR,XPTI)
      CNUM = CDEXP(CXPT)
      CS4 = CS1 + CS2
      CS5 = CS1 + CS3
      CALL CPOWER (CS4,CS6,5.D-1,1,NCALL)
      CALL CPOWER (CS5,CS7,5.D-1,2,NCALL)
      CDENOM = CS6 + CS7
      CRESLT = CNUM/CDENOM
      U = DREAL(CRESLT)
      V = DIMAG(CRESLT)
      RETURN
      END
```

AL = 1.25D-1

BE = -1.25D-1

T = 8.D+0

AA = 0.3

$X_P$ = -4.52

$Y_P$ = 2.2

MMMMM = 23

LOSC = 5

SUM = 1.17209D-1

SUMA = 6.50880D-1

TIME = 0.13801D+0

ANALYTICAL = 1.17209D-1

Case 63.

$$\mathcal{L}^{-1}\left[\frac{1}{s\sqrt{s+a}}\right] = \frac{\operatorname{erf}\sqrt{at}}{\sqrt{a}}$$

```
      SUBROUTINE VALUE (X, Y, T, AL, U, V, MMMMM, NCALL)
C     THIS IS TABLE ENTRY NUMBER 63 OF LAPLACE TRANSFORMS BY SPIEGEL
      IMPLICIT REAL*8 (A,B,D-H,O-Z)
      IMPLICIT COMPLEX*16 (C)
      DIMENSION NCALL(5)
      MMMMM = MMMMM + 1
      ZERO = 0.D+0
      CS1 = DCMPLX(X,Y)
      CS2 = DCMPLX(AL,ZERO)
      XPTR = X*T
      XPTI = Y*T
      CXPT = DCMPLX(XPTR,XPTI)
      CNUM = CDEXP(CXPT)
      CS3 = CS1 + CS2
      CALL CPOWER (CS3,CS4,5.D-1,1,NCALL)
      CDENOM = CS1*CS4
      CRESLT = CNUM/CDENOM
      U = DREAL(CRESLT)
      V = DIMAG(CRESLT)
      RETURN
      END
```

AL = 1.25D-1

T = 8.D+0

AA = 0.2

$X_F$ = -4.64

$y_F$ = 2.2

MMMMM = 23

LOSC = 5

SUM = 2.38352D+0

SUMA = 2.64736D+0

TIME = 0.85202D-1

ANALYTICAL = 2.38352D+0

Case 64.

$$\mathcal{L}^{-1}\left[\frac{1}{\sqrt{s}\,(s-a)}\right] = \frac{e^{at}\,\operatorname{erf}\sqrt{at}}{\sqrt{a}}$$

```
      SUBROUTINE VALUE (X, Y, T, AL, U, V, MMMMM, NCALL)
C     THIS IS TABLE ENTRY NUMBER 64 OF LAPLACE TRANSFORMS BY SPIEGEL
      IMPLICIT REAL*8 (A,B,D-H,O-Z)
      IMPLICIT COMPLEX*16 (C)
      DIMENSION NCALL(5)
      MMMMM = MMMMM + 1
      ZERO = 0.D+0
      CS1 = DCMPLX(X,Y)
      CS2 = DCMPLX(AL,ZERO)
      XPTR = X*T
      XPTI = Y*T
      CXPT = DCMPLX(XPTR,XPTI)
      CNUM = CDEXP(CXPT)
      CALL CPOWER (CS1,CS3,5.D-1,1,NCALL)
      CS4 = CS1 - CS2
      CDENOM = CS3*CS4
      CRESLT = CNUM/CDENOM
      U = DREAL(CRESLT)
      V = DIMAG(CRESLT)
      RETURN
      END
```

AL = 1.25D-1

T = 8.D+0

AA = 0.4

$X_F$ = -4.44

$y_F$ = 2.2

MMMMM = 23

LOSC = 5

SUM = 6.47907D+0

SUMA = 8.56802D+0

TIME = 0.97552D-1

ANALYTICAL = 6.47907D+0

Case 65.

$$\mathcal{L}^{-1}\left[\frac{1}{\sqrt{s-a}+b}\right] = e^{at}\left\{\frac{1}{\sqrt{\pi t}} - b\,e^{b^2 t}\,\mathrm{erfc}(b\sqrt{t})\right\}$$

```
      SUBROUTINE VALUE (X, Y, T, AL, BE, U, V, MMMMM, NCALL)
C     THIS IS TABLE ENTRY NUMBER 65 OF LAPLACE TRANSFORMS BY SPIEGEL
      IMPLICIT REAL*8 (A,B,D-H,O-Z)
      IMPLICIT COMPLEX*16 (C)
      DIMENSION NCALL(5)
      MMMMM = MMMMM + 1
      ZERO = 0.D+0
      CS1 = DCMPLX(X,Y)
      CS2 = DCMPLX(AL,ZERO)
      CS3 = DCMPLX(BE,ZERO)
      XPTR = X*T
      XPTI = Y*T
      CXPT = DCMPLX(XPTR,XPTI)
      CNUM = CDEXP(CXPT)
      CS4 = CS1 - CS2
      CALL CPOWER (CS4,CS5,5.D-1,1,NCALL)
      CDENOM = CS3 + CS5
      CRESLT = CNUM/CDENOM
      U = DREAL(CRESLT)
      V = DIMAG(CRESLT)
      RETURN
      END
```

AL = 1.25D-1

BE = 3.535533906D-1

T = 8.D+0

AA = 0.3

$X_p$ = -4.54

$y_p$ = 2.2

MMMMM = 23

LOSC = 5

SUM = 1.31286D-1

SUMA = 9.05849D-1

TIME = 9.0116D-2

ANALYTICAL = 1.31286D-1

Case 66.

$$\mathcal{L}^{-1}\left[\frac{1}{\sqrt{s^2+a^2}}\right] = J_0(at)$$

```
      SUBROUTINE VALUE (X, Y, T, AL, U, V, MMMMM, NCALL)
C     THIS IS TABLE ENTRY NUMBER 66 OF LAPLACE TRANSFORMS BY SPIEGEL
      IMPLICIT REAL*8 (A,B,D-H,O-Z)
      IMPLICIT COMPLEX*16 (C)
      DIMENSION NCALL(5)
      MMMMM = MMMMM + 1
      ZERO = 0.D+0
      CS1 = DCMPLX(X,Y)
      CS2 = DCMPLX(AL,ZERO)
      XPTR = X*T
      XPTI = Y*T
      CXPT = DCMPLX(XPTR,XPTI)
      CNUM = CDEXP(CXPT)
      CS3 = (CS1*CS1) + (CS2*CS2)
      CALL CPOWER (CS3,CDENOM,5.D-1,1,NCALL)
      CRESLT = CNUM/CDENOM
      U = DREAL(CRESLT)
      V = DIMAG(CRESLT)
      RETURN
      END
```

AL = 1.25D-1

T = 8.D+0

AA = 0.2

$X_F$ = -4.64

$y_F$ = 2.2

MMMMM = 23

LOSC = 5

SUM = 7.65198D-1

SUMA = 1.22916D+0

TIME = 0.91286D-1

ANALYTICAL = 7.65198D-1

Case 67.

$$\mathcal{L}^{-1}\left[\frac{1}{\sqrt{s^2 - a^2}}\right] = I_0(at)$$

```
      SUBROUTINE VALUE (X, Y, T, AL, U, V, MMMMM, NCALL)
C     THIS IS TABLE ENTRY NUMBER 67 OF LAPLACE TRANSFORMS BY SPIEGEL
      IMPLICIT REAL*8 (A,B,D-H,O-Z)
      IMPLICIT COMPLEX*16 (C)
      DIMENSION NCALL(5)
      MMMMM = MMMMM + 1
      ZERO = 0.D+0
      CS1 = DCMPLX(X,Y)
      CS2 = DCMPLX(AL,ZERO)
      XPTR = X*T
      XPTI = Y*T
      CXPT = DCMPLX(XPTR,XPTI)
      CNUM = CDEXP(CXPT)
      CS3 = (CS1*CS1) - (CS2*CS2)
      CALL CPOWER (CS3,CDENOM,5.D-1,1,NCALL)
      CRESLT = CNUM/CDENOM
      U = DREAL(CRESLT)
      V = DIMAG(CRESLT)
      RETURN
      END
```

AL = 1.25D-1

T = 8.D+0

AA = 0.3

$X_F$ = -4.99

$y_F$ = 2.3

MMMMM = 24

LOSC = 5

SUM = 1.26607D+0

SUMA = 2.31076D+0

TIME = 0.10291D+0

ANALYTICAL = 1.26607D+0

Case 68. $\mathcal{L}^{-1}\left[\frac{(\sqrt{s^2+a^2}-s)^n}{\sqrt{s^2+a^2}} \quad n>-1\right] = a^n J_n(at)$

```
      SUBROUTINE VALUE (X, Y, T, AL, NPOWER, U, V, MMMMM, NCALL)
C     THIS IS TABLE ENTRY NUMBER 68 OF LAPLACE TRANSFORMS BY SPIEGEL
      IMPLICIT REAL*8 (A,B,D-H,O-Z)
      IMPLICIT COMPLEX*16 (C)
      DIMENSION NCALL(5)
      MMMMM = MMMMM + 1
      ZERO = 0.D+0
      CS1 = DCMPLX(X,Y)
      CS2 = DCMPLX(AL,ZERO)
      XPTR = X*T
      XPTI = Y*T
      CXPT = DCMPLX(XPTR,XPTI)
      CS3 = (CS1*CS1) + (CS2*CS2)
      CALL CPOWER (CS3,CDENOM,5.D-1,1,NCALL)
      CS4 = (CDENOM - CS1)**NPOWER
      CNUM = CS4*CDEXP(CXPT)
      CRESLT = CNUM/CDENOM
      U = DREAL(CRESLT)
      V = DIMAG(CRESLT)
      RETURN
      END
```

AL = 1.25D-1

NPOWER = 3

T = 8.D+0

AA = 0.4

$X_P$ = -2.16

$y_P$ = 1.6

MMMMM = 17

LOSC = 1

SUM = 3.82096D-5

SUMA = 3.82096D-5

TIME = 0.81224D-1

ANALYTICAL = 3.82090D-1

Case 69. $\mathcal{L}^{-1}\left[\frac{(s-\sqrt{s^2-a^2})^n}{\sqrt{s^2-a^2}} \quad n > -1\right] = a^n I_n(at)$

```
      SUBROUTINE VALUE (X, Y, T, AL, NPOWER, U, V, MMMMM, NCALL)
C     THIS IS TABLE ENTRY NUMBER 69 OF LAPLACE TRANSFORMS BY SPIEGEL
      IMPLICIT REAL*8 (A,B,D-H,O-Z)
      IMPLICIT COMPLEX*16 (C)
      DIMENSION NCALL(5)
      MMMMM = MMMMM + 1
      ZERO = 0.D+0
      CS1 = DCMPLX(X,Y)
      CS2 = DCMPLX(AL,ZERO)
      XPTR = X*T
      XPTI = Y*T
      CXPT = DCMPLX(XPTR,XPTI)
      CS3 = (CS1*CS1) - (CS2*CS2)
      CALL CPOWER (CS3,CDENOM,5.D-1,1,NCALL)
      CS4 = (CS1 - CDENOM)**NPOWER
      CNUM = CS4*CDEXP(CXPT)
      CRESLT = CNUM/CDENOM
      U = DREAL(CRESLT)
      V = DIMAG(CRESLT)
      RETURN
      END
```

AL = 1.25D-1

NPOWER = 3

T = 8.D+0

AA = 0.4

$X_F$ = -2.16

$y_F$ = 1.6

MMMMM = 17

LOSC = 1

SUM = 4.32979D-5

SUMA = 4.32979D-5

TIME = 0.74672D-1

ANALYTICAL = 4.32977D-5

Case 70.

$$\mathcal{L}^{-1}\left[\frac{e^{b(s-\sqrt{s^2+a^2})}}{\sqrt{s^2+a^2}}\right] = J_0(a\sqrt{t(t+2b)})$$

```
      SUBROUTINE VALUE (X, Y, T, AL, BE, U, V, MMMMM, NCALL)
C     THIS IS TABLE ENTRY NUMBER 70 OF LAPLACE TRANSFORMS BY SPIEGEL
      IMPLICIT REAL*8 (A,B,D-H,O-Z)
      IMPLICIT COMPLEX*16 (C)
      DIMENSION NCALL(5)
      MMMMM = MMMMM + 1
      ZERO = 0.D+0
      CS1 = DCMPLX(X,Y)
      CS2 = DCMPLX(AL,ZERO)
      CS3 = DCMPLX(BE,ZERO)
      XPTR = X*T
      XPTI = Y*T
      CXPT = DCMPLX(XPTR,XPTI)
      CS4 = (CS1*CS1) + (CS2*CS2)
      CALL CPOWER (CS4,CDENOM,5.D-1,1,NCALL)
      CS5 = (CS1 - CDENOM)*CS3
      CNUM = CDEXP(CS5)*CDEXP(CXPT)
      CRESLT = CNUM/CDENOM
      U = DREAL(CRESLT)
      V = DIMAG(CRESLT)
      RETURN
      END
```

AL = 1.25D-1

BE = -1.25D-1

T = 8.D+0

AA = 0.2

$X_F$ = -4.64

$y_F$ = 2.2

MMMMM = 23

LOSC = 5

* SUM = 7.72088D-1

SUMA = 1.23485D+0

TIME = 0.82914D-1

ANALYTICAL = 7.72088D-1

Case 71.

$$\mathcal{L}^{-1}\left[\frac{e^{-b\sqrt{s^2+a^2}}}{\sqrt{s^2+a^2}}\right] = \begin{cases} J_0(a\sqrt{t^2-b^2}) & t > b \\ 0 & t < b \end{cases}$$

```
      SUBROUTINE VALUE (X, Y, T, AL, BE, U, V, MMMMM, NCALL)
C     THIS IS TABLE ENTRY NUMBER 71 OF LAPLACE TRANSFORMS BY SPIEGEL
      IMPLICIT REAL*8 (A,B,D-H,O-Z)
      IMPLICIT COMPLEX*16 (C)
      DIMENSION NCALL(5)
      MMMMM = MMMMM + 1
      ZERO = 0.D+0
      CS1 = DCMPLX(X, Y)
      CS2 = DCMPLX(AL, ZERO)
      CS3 = DCMPLX(-BE, ZERO)
      XPTR = X*T
      XPTI = Y*T
      CXPT = DCMPLX(XPTR,XPTI)
      CS4 = (CS1*CS1) + (CS2*CS2)
      CALL CPOWER (CS4,CDENOM,5.D-1,1,NCALL)
      CNUM = CDEXP(CDENOM*CS3)*CDEXP(CXPT)
      CRESLT = CNUM/CDENOM
      U = DREAL(CRESLT)
      V = DIMAG(CRESLT)
      RETURN
      END
```

AL = 1.25D-1

BE = -1.25D-1

T = 8.D+0

AA = 0.2

$X_F$ = -4.64

$y_F$ = 2.2

MMMMM = 23

LOSC = 5

SUM = 7.65252D-1

SUMA = 1.24690D+0

TIME = 0.86658D-1

ANALYTICAL = 7.65252D-1

Case 72.

$$\mathcal{L}^{-1}\left[\frac{1}{(s^2+a^2)^{3/2}}\right] = \frac{tJ_1(at)}{a}$$

```
      SUBROUTINE VALUE (X, Y, T, AL, U, V, MMMMM, NCALL)
C     THIS IS TABLE ENTRY NUMBER 72 OF LAPLACE TRANSFORMS BY SPIEGEL
      IMPLICIT REAL*8 (A,B,D-H,O-Z)
      IMPLICIT COMPLEX*16 (C)
      DIMENSION NCALL(5)
      MMMMM = MMMMM + 1
      ZERO = 0.D+0
      XPTR = X*T
      XPTI = Y*T
      CXPT = DCMPLX(XPTR,XPTI)
      CNUM = CDEXP(CXPT)
      XREAL = (X*X) - (Y*Y) + (AL*AL)
      XIMAG = 2.D+0*X*Y
      CS1 = DCMPLX(XREAL,XIMAG)
      CS2 = CS1*CS1*CS1
      CALL CPOWER (CS2, CDENOM, 5.D-1, 1, NCALL)
      CRESLT = CNUM/CDENOM
      U = DREAL(CRESLT)
      V = DIMAG(CRESLT)
      RETURN
      END
```

AL = 1.25D-1

T = 8.D+0

AA = 0.4

$x_F$ = -4.44

$y_P$ = 2.2

MMMMM = 23

LOSC = 3

SUM = 2.81632D+1

SUMA = 2.81715D+1

TIME = 0.11476D+0

ANALYTICAL = 2.81632D+1

Case 73.

$$\mathcal{L}^{-1}\left[\frac{s}{(s^2+a^2)^{3/2}}\right] = tJ_0(at)$$

```
      SUBROUTINE VALUE (X, Y, T, AL, U, V, MMMMM, NCALL)
C     THIS IS TABLE ENTRY NUMBER 73 OF LAPLACE TRANSFORMS BY SPIEGEL
      IMPLICIT REAL*8 (A,B,D-H,O-Z)
      IMPLICIT COMPLEX*16 (C)
      DIMENSION NCALL(5)
      MMMMM = MMMMM + 1
      ZERO = 0.D+0
      CS1 = DCMPLX(X,Y)
      XPTR = X*T
      XPTI = Y*T
      CXPT = DCMPLX(XPTR,XPTI)
      CS2 = CDEXP(CXPT)
      CNUM = CS1*CS2
      XREAL = (X*X) - (Y*Y) + (AL*AL)
      XIMAG = 2.D+0*X*Y
      CS3 = DCMPLX(XREAL,XIMAG)
      CS4 = CS3*CS3*CS3
      CALL CPOWER (CS4, CDENOM, 5.D-1, 1, NCALL)
      CRESLT = CNUM/CDENOM
      U = DREAL(CRESLT)
      V = DIMAG(CRESLT)
      RETURN
      END
```

AL = 1.25D-1

T = 8.D+0

AA = 0.4

$X_F$ = -4.89

$y_F$ = 2.3

MMMMM = 24

LOSC = 4

SUM = 6.12158D+0

SUMA = 8.34998D+0

TIME = 0.12620D+0

ANALYTICAL = 6.12158D+0

Case 74.

$$\mathcal{L}^{-1}\left[\frac{s^2}{(s^2+a^2)^{3/2}}\right] = J_0(at) - at\,J_1(at)$$

```
      SUBROUTINE VALUE (X, Y, T, AL, U, V, MMMMM, NCALL)
C     THIS IS TABLE ENTRY NUMBER 74 OF LAPLACE TRANSFORMS BY SPIEGEL
      IMPLICIT REAL*8 (A,B,D-H,O-Z)
      IMPLICIT COMPLEX*16 (C)
      DIMENSION NCALL(5)
      MMMMM = MMMMM + 1
      ZERO = 0.D+0
      CS1 = DCMPLX(X,Y)
      XPTR = X*T
      XPTI = Y*T
      CXPT = DCMPLX(XPTR,XPTI)
      CS2 = CDEXP(CXPT)
      CS3 = CS1*CS1
      CNUM = CS2*CS3
      XREAL = (X*X) - (Y*Y) + (AL*AL)
      XIMAG = 2.D+0*X*Y
      CS4 = DCMPLX(XREAL,XIMAG)
      CS5 = CS4*CS4*CS4
      CALL CPOWER (CS5, CDENOM, 5.D-1, 1, NCALL)
      CRESLT = CNUM/CDENOM
      U = DREAL(CRESLT)
      V = DIMAG(CRESLT)
      RETURN
      END
```

AL = 1.25D-1

T = 8.D+0

AA =0.4

$X_F$ = -4.89

$y_F$ = 2.3

MMMMM = 24

LOSC = 5

SUM = 3.25147D-1

SUMA = 3.72547D+0

TIME = 0.10772D+0

ANALYTICAL = 3.25147D-1

Case 75.

$$\mathcal{L}^{-1}\left[\frac{1}{(s^2 - a^2)^{3/2}}\right] = \frac{t I_1(at)}{a}$$

```
      SUBROUTINE VALUE (X, Y, T, AL, U, V, MMMMM, NCALL)
C     THIS IS TABLE ENTRY NUMBER 75 OF LAPLACE TRANSFORMS BY SPIEGEL
      IMPLICIT REAL*8 (A,B,D-H,O-Z)
      IMPLICIT COMPLEX*16 (C)
      DIMENSION NCALL(5)
      MMMMM = MMMMM + 1
      ZERO = 0.D+0
      XPTR = X*T
      XPTI = Y*T
      CXPT = DCMPLX(XPTR,XPTI)
      CNUM = CDEXP(CXPT)
      XREAL = (X*X) - (Y*Y) - (AL*AL)
      XIMAG = 2.D+0*X*Y
      CS1 = DCMPLX(XREAL,XIMAG)
      CS2 = CS1*CS1*CS1
      CALL CPOWER (CS2, CDENOM, 5.D-1, 1, NCALL)
      CRESLT = CNUM/CDENOM
      U = DREAL(CRESLT)
      V = DIMAG(CRESLT)
      RETURN
      END
```

AL = 1.25D-1

T = 8.D+0

AA = 0.5

$x_F$ = -4.34

$y_F$ = 2.2

MMMMM = 23

LOSC = 3

SUM = 3.61702D+1

SUMA = 3.63846D+1

TIME = 0.11089D+0

ANALYTICAL = 3.61702D+1

Case 76.

$$\mathcal{L}^{-1}\left[\frac{s}{(s^2-a^2)^{3/2}}\right] = t\,I_0(at)$$

```
      SUBROUTINE VALUE (X, Y, T, AL, U, V, MMMMM, NCALL)
C     THIS IS TABLE ENTRY NUMBER 76 OF LAPLACE TRANSFORMS BY SPIEGEL
      IMPLICIT REAL*8 (A,B,D-H,O-Z)
      IMPLICIT COMPLEX*16 (C)
      DIMENSION NCALL(5)
      MMMMM = MMMMM + 1
      ZERO = 0.D+0
      CS1 = DCMPLX(X, Y)
      XPTR = X*T
      XPTI = Y*T
      CXPT = DCMPLX(XPTR,XPTI)
      CS2 = CDEXP(CXPT)
      CNUM = CS1*CS2
      XREAL = (X*X) - (Y*Y) - (AL*AL)
      XIMAG = 2.D+0*X*Y
      CS3 = DCMPLX(XREAL,XIMAG)
      CS4 = CS3*CS3*CS3
      CALL CPOWER (CS4, CDENOM, 5.D-1, 1, NCALL)
      CRESLT = CNUM/CDENOM
      U = DREAL(CRESLT)
      V = DIMAG(CRESLT)
      RETURN
      END
```

AL = 1.25D-1

T = 8.D+0

AA = 0.5

$X_F$ = -4.79

$y_F$ = 2.3

MMMMM = 24

LOSC = 4

SUM = 1.01285D+1

SUMA = 1.52218D+1

TIME = 0.13250D+0

ANALYTICAL = 1.01285D+1

Case 77.

$$\mathcal{L}^{-1}\left[\frac{s^2}{(s^2-a^2)^{3/2}}\right] = I_0(at) + at\,I_1(at)$$

```
      SUBROUTINE VALUE (X, Y, T, AL, U, V, MMMMM, NCALL)
C     THIS IS TABLE ENTRY NUMBER 77 OF LAPLACE TRANSFORMS BY SPIEGEL
      IMPLICIT REAL*8 (A,B,D-H,O-Z)
      IMPLICIT COMPLEX*16 (C)
      DIMENSION NCALL(5)
      MMMMM = MMMMM + 1
      ZERO = 0.D+0
      CS1 = DCMPLX(X, Y)
      XPTR = X*T
      XPTI = Y*T
      CXPT = DCMPLX(XPTR,XPTI)
      CS2 = CDEXP(CXPT)
      CS3 = CS1*CS1
      CNUM = CS2*CS3
      XREAL = (X*X) - (Y*Y) - (AL*AL)
      XIMAG = 2.D+0*X*Y
      CS4 = DCMPLX(XREAL,XIMAG)
      CS5 = CS4*CS4*CS4
      CALL CPOWER (CS5, CDENOM, 5.D-1, 1, NCALL)
      CRESLT = CNUM/CDENOM
      U = DREAL(CRESLT)
      V = DIMAG(CRESLT)
      RETURN
      END
```

AL = 1.25D-1

T = 8.D+0

AA = 0.5

$X_F$ = -4.79

$y_F$ = 2.3

MMMMM = 24

LOSC = 5

SUM = 1.83122D+0

SUMA = 7.71705D+0

TIME = 0.11807D+0

ANALYTICAL = 1.83122D+0

Case 78

$$\mathcal{L}^{-1}\left[\frac{e^{-a/s}}{\sqrt{s}}\right] = \frac{\cos 2\sqrt{at}}{\sqrt{\pi t}}$$

```
      SUBROUTINE VALUE (X, Y, T, AL, U, V, MMMMM, NCALL)
C     THIS IS TABLE ENTRY NUMBER 81 OF LAPLACE TRANSFORMS BY SPIEGEL
      IMPLICIT REAL*8 (A,B,D-H,O-Z)
      IMPLICIT COMPLEX*16 (C)
      DIMENSION NCALL(5)
      MMMMM = MMMMM + 1
      ZERO = 0.D+0
      CS1 = DCMPLX(X,Y)
      CS2 = DCMPLX(AL,ZERO)
      CS3 = -CS2/CS1
      XPTR = X*T
      XPTI = Y*T
      CXPT = DCMPLX(XPTR,XPTI)
      CNUM = CDEXP(CS3)*CDEXP(CXPT)
      CALL CPOWER (CS1,CDENOM,5.D-1,1,NCALL)
      CRESLT = CNUM/CDENOM
      U = DREAL(CRESLT)
      V = DIMAG(CRESLT)
      RETURN
      END
```

AL = 1.25D-1

T = 8.D+0

AA = 0.2

$x_p$ = -4.64

$y_p$ = 2.2

MMMMM = 23

LOSC = 5

SUM = -8.30093D-2

SUMA = 4.47194D-1

TIME = 0.96148D-1

ANALYTICAL = -8.30093D-2

Case 79

$$\mathcal{L}^{-1}\left[\frac{e^{-a/s}}{s^{3/2}}\right] = \frac{\sin 2\sqrt{at}}{\sqrt{\pi a}}$$

```
      SUBROUTINE VALUE (X, Y, T, AL, U, V, MMMMM, NCALL)
C     THIS IS TABLE ENTRY NUMBER 82 OF LAPLACE TRANSFORMS BY SPIEGEL
      IMPLICIT REAL*8 (A,B,D-H,O-Z)
      IMPLICIT COMPLEX*16 (C)
      DIMENSION NCALL(5)
      MMMMM = MMMMM + 1
      ZERO = 0.D+0
      CS1 = DCMPLX(X,Y)
      CS2 = DCMPLX(AL,ZERO)
      CS3 = -CS2/CS1
      XPTR = X*T
      XPTI = Y*T
      CXPT = DCMPLX(XPTR,XPTI)
      CNUM = CDEXP(CS3)*CDEXP(CXPT)
      CS4 = CS1*CS1*CS1
      CALL CPOWER (CS4,CDENOM,5.D-1,1,NCALL)
      CRESLT = CNUM/CDENOM
      U = DREAL(CRESLT)
      V = DIMAG(CRESLT)
      RETURN
      END
```

AL = 1.25D-1

T = 8.D+0

AA = 0.2

$X_F$ = -4.64

$y_F$ = 2.2

MMMMM = 23

LOSC = 5

SUM = 1.45103D+0

SUMA = 1.82676D+0

TIME = 0.10018D+0

ANALYTICAL = 1.45103D+0

Case 80

$$\mathcal{L}^{-1}\left[\frac{e^{-a/s}}{s^{n+1}} \quad n > -1\right] = \left(\frac{t}{a}\right)^{n/2} J_n(2\sqrt{at})$$

```
      SUBROUTINE VALUE (X, Y, T, AL, NPOWER, U, V, MMMMM, NCALL)
C     THIS IS TABLE ENTRY NUMBER 83 OF LAPLACE TRANSFORMS BY SPIEGEL
      IMPLICIT REAL*8 (A,B,D-H,O-Z)
      IMPLICIT COMPLEX*16 (C)
      DIMENSION NCALL(5)
      MMMMM = MMMMM + 1
        ZERO = 0.D+0
      CS1 = DCMPLX(X, Y)
      CS2 = DCMPLX(AL, ZERO)
      XPTR = X*T
      XPTI = Y*T
      CXPT = DCMPLX(XPTR,XPTI)
      CNUM = CDEXP(CS3)*CDEXP(CXPT)
      CDENOM = CS1**(NPOWER + 1)
      CRESLT = CNUM/CDENOM
      U = DREAL(CRESLT)
      V = DIMAG(CRESLT)
      RETURN
      END
```

AL = 1.25D-1

NPOWER = 3

T = 8.D+0

AA = 0.3

$X_P$ = -4.11

$y_P$ = 2.1

MMMMM = 22

LOSC = 4

SUM = 6.60189D+1

SUMA = 6.60189D+1

TIME = 6.9758D-1

ANALYTICAL = 6.60189D+1

Case 81.

$$\mathcal{L}^{-1}\left[\frac{e^{-a\sqrt{s}}}{\sqrt{s}}\right] = \frac{e^{-a^2/4t}}{\sqrt{\pi t}}$$

```
      SUBROUTINE VALUE (X, Y, T, AL, U, V, MMMMM, NCALL)
C     THIS IS TABLE ENTRY NUMBER 84 OF LAPLACE TRANSFORMS BY SPIEGEL
      IMPLICIT REAL*8 (A,B,D-H,O-Z)
      IMPLICIT COMPLEX*16 (C)
      DIMENSION NCALL(5)
      MMMMM = MMMMM + 1
        ZERO = 0.D+0
      CS1 = DCMPLX(X,Y)
      CS2 = DCMPLX(AL,ZERO)
      XPTR = X*T
      XPTI = Y*T
      CXPT = DCMPLX(XPTR,XPTI)
      CALL CPOWER (CS1,CDENOM,5.D-1,1,NCALL)
      CS3 = -CS2*CDENOM
      CNUM = CDEXP(CS3)*CDEXP(CXPT)
      CRESLT = CNUM/CDENOM
      U = DREAL(CRESLT)
      V = DIMAG(CRESLT)
      RETURN
      END
```

AL = 1.25D-1

T = 8.D+0

AA = 0.2

$X_F$ = -4.64

$y_F$ = 2.2

MMMMM = 23

LOSC = 5

SUM = 1.99374D-1

SUMA = 6.27444D-1

TIME = 0.93470D-1

ANALYTICAL = 1.99374D-1

Case 82.

$$\mathcal{L}^{-1}\left[e^{-a\sqrt{s}}\right] = \frac{a}{2\sqrt{\pi t^3}}e^{-a^2/4t}$$

```
      SUBROUTINE VALUE (X, Y, T, AL, U, V, MMMMM, NCALL)
C     THIS IS TABLE ENTRY NUMBER 85 OF LAPLACE TRANSFORMS BY SPIEGEL
      IMPLICIT REAL*8 (A,B,D-H,O-Z)
      IMPLICIT COMPLEX*16 (C)
      DIMENSION NCALL(5)
      MMMMM = MMMMM + 1
         ZERO = 0.D+0
      CS1 = DCMPLX(X,Y)
      CS2 = DCMPLX(AL,ZERO)
      XPTR = X*T
      XPTI = Y*T
      CXPT = DCMPLX(XPTR,XPTI)
      CALL CPOWER (CS1,CS3,5.D-1,1,NCALL)
      CS4 = -CS2*CS3
      CRESLT = CDEXP(CS4)*CDEXP(CXPT)
      U = DREAL(CRESLT)
      V = DIMAG(CRESLT)
      RETURN
      END
```

AL = 1.25D-1

T = 8.D+0

AA = 0.2

$x_F$ = -5.09

$y_F$ = 2.3

MMMMM = 24

LOSC = 6

SUM = 1.55761D-3

SUMA = 3.03773D-1

TIME = 0.10197D+0

ANALYTICAL = 1.55761D-3

Case 83.

$$\mathcal{L}^{-1}\left[\frac{1-e^{-a\sqrt{s}}}{s}\right] = \operatorname{erf}(a/2\sqrt{t})$$

```
      SUBROUTINE VALUE (X, Y, T, AL, U, V, MMMMM, NCALL)
C     THIS IS TABLE ENTRY NUMBER 86 OF LAPLACE TRANSFORMS BY SPIEGEL
      IMPLICIT REAL*8 (A,B,D-H,O-Z)
      IMPLICIT COMPLEX*16 (C)
      DIMENSION NCALL(5)
      MMMMM = MMMMM + 1
          ZERO = 0.D+0
      ONE = 1.D+0
      CS1 = DCMPLX(X,Y)
      CS2 = DCMPLX(AL,ZERO)
      CS3 = DCMPLX(ONE,ZERO)
      XPTR = X*T
      XPTI = Y*T
      CXPT = DCMPLX(XPTR,XPTI)
      CALL CPOWER (CS1,CS4,5.D-1,1,NCALL)
      CNUM = (CS3 - CDEXP(-CS2*CS4))*CDEXP(CXPT)
      CDENOM = CS1
      CRESLT = CNUM/CDENOM
      U = DREAL(CRESLT)
      V = DIMAG(CRESLT)
      RETURN
      END
```

AL = 1.25D-1

T = 8.D+0

AA = 0.2

$X_F$ = -4.64

$y_F$ = 2.2

MMMMM = 23

LOSC = 5

SUM = 2.49298D-2

SUMA = 8.07480D-2

TIME = 0.93158D-1

ANALYTICAL = 2.49292D-2

Case 84.

$$\mathcal{L}^{-1}\left[\frac{e^{-a\sqrt{s}}}{s}\right] = \operatorname{erfc}(a/2\sqrt{t})$$

```
      SUBROUTINE VALUE (X, Y, T, AL, U, V, MMMMM, NCALL)
C     THIS IS TABLE ENTRY NUMBER 87 OF LAPLACE TRANSFORMS BY SPIEGEL
      IMPLICIT REAL*8 (A,B,D-H,O-Z)
      IMPLICIT COMPLEX*16 (C)
      DIMENSION NCALL(5)
      MMMMM = MMMMM + 1
            ZERO = 0.D+0
      CS1 = DCMPLX(X,Y)
      CS2 = DCMPLX(AL,ZERO)
      XPTR = X*T
      XPTI = Y*T
      CXPT = DCMPLX(XPTR,XPTI)
      CALL CPOWER (CS1,CS4,5.D-1,1,NCALL)
      CNUM = CDEXP(-CS2*CS4)*CDEXP(CXPT)
      CDENOM = CS1
      CRESLT = CNUM/CDENOM
      U = DREAL(CRESLT)
      V = DIMAG(CRESLT)
      RETURN
      END
```

AL = 1.25D-1

T = 8.D+0

AA = 0.2

$x_P$ = -4.64

$y_P$ = 2.2

MMMMM = 23

LOSC = 5

SUM = 9.75070D-1

SUMA = 1.35850D+0

TIME = 0.99658D-1

ANALYTICAL = 9.75071D-1

Case 85.

$$\mathcal{L}^{-1}\left[\frac{e^{-a\sqrt{s}}}{\sqrt{s}(\sqrt{s}+b)}\right] = e^{b(bt+a)}\operatorname{erfc}\left(b\sqrt{t}+\frac{a}{2\sqrt{t}}\right)$$

```
      SUBROUTINE VALUE (X, Y, T, AL, BE, U, V, MMMMM, NCALL)
C     THIS IS TABLE ENTRY NUMBER 88 OF LAPLACE TRANSFORMS BY SPIEGEL
      IMPLICIT REAL*8 (A,B,C-H,O-Z)
      IMPLICIT COMPLEX*16 (C)
      DIMENSION NCALL(5)
      MMMMM = MMMMM + 1
              ZERO = 0.D+0
      CS1 = DCMPLX(X,Y)
      CS2 = DCMPLX(AL,ZERO)
      XPTR = X*T
      XPTI = Y*T
      CXPT = DCMPLX(XPTR,XPTI)
      CS3 = DCMPLX(BE,ZERO)
      CALL CPOWER (CS1,CS4,5.D-1,1,NCALL)
      CNUM = CDEXP(-CS2*CS4)*CDEXP(CXPT)
      CS5 = CS3 + CS4
      CDENOM = CS4*CS5
      CRESLT = CNUM/CDENOM
      U = DREAL(CRESLT)
      V = DIMAG(CRESLT)
      RETURN
      END
```

AL = 1.25D-1

BE = 3.457408906D-1

T = 8.D+0

AA = 0.2

$X_P$ = -4.64

$y_P$ = 2.2

MMMMM = 23

LOSC = 3

SUM = 4.27375D-1

SUMA = 7.5575D-1

TIME = 9.9294D-2

ANALYTICAL = 4.27375D-1

Case 86.

$$\mathcal{L}^{-1}\left[\ln\left(\frac{s+a}{s+b}\right)\right] = \frac{e^{-bt}-e^{-at}}{t}$$

```
      SUBROUTINE VALUE (X,Y,T,AL,BE,U,V,MMMMM,NCALL,MCALL)
C     THIS IS TABLE ENTRY NUMBER 90 OF LAPLACE TRANSFORMS BY SPIEGEL
      IMPLICIT REAL*8 (A,B,D-H,O-Z)
      IMPLICIT COMPLEX*16 (C)
      DIMENSION NCALL(5), MCALL(5)
      MMMMM = MMMMM + 1
      ZERO = 0.D+0
      CS1 = DCMPLX(X,Y)
      CS2 = DCMPLX(AL,ZERO)
      CS3 = DCMPLX(BE,ZERO)
      XPTR = X*T
      XPTI = Y*T
      CXPT = DCMPLX(XPTR,XPTI)
      CS4 = CS1 + CS2
      CS5 = CS1 + CS3
      CS6 = CS4/CS5
      CALL CLOG (CS6, CS7, 1, MCALL)
      CRESLT = CS7*CDEXP(CXPT)
      U = DREAL(CRESLT)
      V = DIMAG(CRESLT)
      RETURN
      END
```

AL = 1.25D-1

BE = 2.5D-1

T = 8.D+0

AA = 0.2

$X_P$ = -4.21

$Y_P$ = 2.1

MMMMM = 22

LOSC = 5

SUM = -2.90680D-2

SUMA = 1.01743D-1

TIME = 7.0096D-2

ANALYTICAL = -2.90680D-2

Case 87.

$$\mathcal{L}^{-1}\left[\frac{\ln[(s^2+a^2)/a^2]}{2s}\right] = \mathrm{Ci}(at)$$

```
      SUBROUTINE VALUE (X, Y, T, AL, U, V, MMMMM, MCALL)
C     THIS IS TABLE ENTRY NUMBER 91 OF LAPLACE TRANSFORMS BY SPIEGEL
      IMPLICIT REAL*8 (A,B,D-H,O-Z)
      IMPLICIT COMPLEX*16 (C)
      DIMENSION MCALL (5)
      MMMMM = MMMMM + 1
      ZERO = 0.D+0
      CS1 = DCMPLX(X,Y)
      CS2 = DCMPLX(AL,ZERO)
      CS3 = ((CS1*CS1) + (CS2*CS2))/(CS2*CS2)
      XPTR = X*T
      XPTI = Y*T
      CXPT = DCMPLX(XPTR,XPTI)
      CALL CLOG (CS3, CS4, 1, MCALL)
      CNUM = CS4*CDEXP(CXPT)
      CS5 = DCMPLX(2.D+0,ZERO)
      CRESLT = CNUM/(CS5*CS1)
      U = DREAL(CRESLT)
      V = DIMAG(CRESLT)
      RETURN
      END
```

AL = 1.25D-1

T = 8.D+0

AA = 0.2

$X_F$ = -5.09

$y_F$ = 2.3

MMMMM = 24

LOSC = 5

SUM = -3.37404D-1

SUMA = 1.27603D+0

TIME = 8.8244D-2

ANALYTICAL = -3.37404D-1

Case 88.

$$\mathcal{L}^{-1}\left[\frac{\ln[(s+a)/a]}{s}\right] = \mathrm{Ei}(at)$$

```
      SUBROUTINE VALUE (X, Y, T, AL, U, V, MMMMM, MCALL)
C     THIS IS TABLE ENTRY NUMBER 92 OF LAPLACE TRANSFORMS BY SPIEGEL
      IMPLICIT REAL*8 (A,B,D-H,O-Z)
      IMPLICIT COMPLEX*16 (C)
      DIMENSION MCALL (5)
      MMMMM = MMMMM + 1
      ZERO = 0.D+0
      CS1 = DCMPLX(X,Y)
      CS2 = DCMPLX(AL,ZERO)
      CS3 = (CS1 + CS2)/CS2
      XPTR = X*T
      XPTI = Y*T
      CXPT = DCMPLX(XPTR,XPTI)
      CALL CLOG (CS3, CS4, 1, MCALL)
      CNUM = CS4*CDEXP(CXPT)
      CRESLT = CNUM/CS1
      U = DREAL(CRESLT)
      V = DIMAG(CRESLT)
      RETURN
      END
```

AL = 1.25D-1

T = 8.D+0

AA = 0.2

$x_F$ = -5.09

$y_F$ = 2.3

MMMMM = 24

LOSC = 5

SUM = 2.19384D-1

SUMA = 1.55094D+0

TIME = 7.2124D-2

ANALYTICAL = 2.19384D-1

Case 89

$$\mathcal{L}^{-1}\left[\frac{\ln s}{s}\right] = -(\ln t + \gamma)$$

$\gamma$ = Euler's constant = .5772156...

```
      SUBROUTINE VALUE (X, Y, T, AL, U, V, MMMMM, MCALL)
C     THIS IS TABLE ENTRY NUMBER 96 OF LAPLACE TRANSFORMS BY SPIEGEL
      IMPLICIT REAL*8 (A,B,D-H,O-Z)
      IMPLICIT COMPLEX*16 (C)
      DIMENSION MCALL (5)
      MMMMM = MMMMM + 1
      ZERO = 0.D+0
      CS1 = DCMPLX(X,Y)
      CALL CLOG (CS1, CS2, 1, MCALL)
      XPTR = X*T
      XPTI = Y*T
      CXPT = DCMPLX(XPTR,XPTI)
      CNUM = CS2*CDEXP(CXPT)
      CRESLT = CNUM/CS1
      U = DREAL(CRESLT)
      V = DIMAG(CRESLT)
      RETURN
      END
```

AL = 1.25D-1

T = 8.D+0

AA = 0.2

$X_P$ = -5.09

$y_F$ = 2.3

MMMMM = 24

LOSC = 5

SUM = -2.65666D+0

SUMA = 2.80606D+0

TIME = 7.7818D-2

ANALYTICAL = -2.65666D+0

Case 90.

$$\mathcal{L}^{-1}\left[\frac{\ln^2 s}{s}\right] = (\ln t + \gamma)^2 - \tfrac{1}{6}\pi^2$$

$\gamma$ = Euler's constant = .5772156...

```
      SUBROUTINE VALUE (X, Y, T, AL, U, V, MMMMM, MCALL)
C     THIS IS TABLE ENTRY NUMBER 97 OF LAPLACE TRANSFORMS BY SPIEGEL
      IMPLICIT REAL*8 (A,B,D-H,O-Z)
      IMPLICIT COMPLEX*16 (C)
      DIMENSION MCALL (5)
      MMMMM = MMMMM + 1
      ZERO = 0.D+0
      CS1 = DCMPLX(X,Y)
      XPTR = X*T
      XPTI = Y*T
      CXPT = DCMPLX(XPTR,XPTI)
      CALL CLOG (CS1, CS2, 1, MCALL)
      CNUM = CS2*CS2*CDEXP(CXPT)
      CRESLT = CNUM/CS1
      U = DREAL(CRESLT)
      V = DIMAG(CRESLT)
      RETURN
      END
```

AL = 1.25D-1

T = 8.D+0

AA = 0.2

$X_P$ = -5.09

$y_P$ = 2.3

MMMMM = 24

LOSC = 4

SUM = 5.41289D+0

SUMA = 5.43447D+0

TIME = 7.7376D-2

ANALYTICAL = 5.41289D+0

Case 91.

$$\mathcal{L}^{-1}\left[\frac{\ln\left[\frac{s+\left[s^2+\alpha^2\right]^{1/2}}{\alpha}\right]}{\left[s^2+\alpha^2\right]^{1/2}}\right] = -\frac{\pi}{2}Y_0(\alpha t)$$

```
      SUBROUTINE VALUE (X,Y,T,AL,BE,U,V,MMMMM,NCALL,MCALL)
C     THIS IS TABLE ENTRY #21 PAGE 265 ROBERTS AND KAUFMAN
      IMPLICIT REAL*8 (A,B,D-H,O-Z)
      IMPLICIT COMPLEX*16 (C)
      DIMENSION NCALL(5), MCALL(5)
      MMMMM = MMMMM + 1
      ZERO = 0.D+0
      CS1 = DCMPLX(X,Y)
      CS2 = DCMPLX(AL,ZERO)
      XPTR = X*T
      XPTI = Y*T
      CXPT = DCMPLX(XPTR,XPTI)
      CS3 = ((CS1*CS1) + (CS2*CS2))
      CALL CPOWER (CS3,CDENOM,5.D-1,1,NCALL)
      CS4 = (CDENOM + CS1)/CS2
      CALL CLOG (CS4, CS5, 1, MCALL)
      CNUM = CS5*CDEXP(CXPT)
      CRESLT = CNUM/CDENOM
      U = DREAL(CRESLT)
      V = DIMAG(CRESLT)
      RETURN
      END
```

AL = 1.25D-1

T = 8.D+0

AA = 0.3

$X_F$ = -4.99

$y_F$ = 2.3

MMMMM = 24

LOSC = 5

SUM = -1.38634D-1

SUMA = 3.92274D+0

TIME = 1.2126D+0

ANALYTICAL = -1.38634D-1

Case 92.

$$\mathcal{L}^{-1}\left[\frac{\ln\left[\dfrac{\left[s^2+a^2\right]^{1/2}+a}{s}\right]}{\left[s^2+a^2\right]^{1/2}}\right] = \frac{\pi}{2} H_0(at)$$

This is known as the Struve function. See ref. (1), page 496.

```
      SUBROUTINE VALUE (X,Y,T,AL,BE,U,V,MMMMM,NCALL,MCALL)
C     THIS IS TABLE ENTRY #12 PAGE 264 ROBERTS AND KAUFMAN
      IMPLICIT REAL*8 (A,B,D-H,O-Z)
      IMPLICIT COMPLEX*16 (C)
      DIMENSION NCALL(5), MCALL(5)
      MMMMM = MMMMM + 1
      ZERO = 0.D+0
      CS1 = DCMPLX(X,Y)
      CS2 = DCMPLX(AL,ZERO)
      XPTR = X*T
      XPTI = Y*T
      CXPT = DCMPLX(XPTR,XPTI)
      CS3 = ((CS1*CS1) + (CS2*CS2))
      CALL CPOWER (CS3,CDENOM,5.D-1,1,NCALL)
      CS4 = (CDENOM + CS2)/CS1
      CALL CLOG (CS4,CS5,1,MCALL)
      CNUM = CS5*CDEXP(CXPT)
      CRESLT = CNUM/CDENOM
      U = DREAL(CRESLT)
      V = DIMAG(CRESLT)
      RETURN
      END
```

AL = 1.25D-1

T = 8.D+0

AA = 0.3

$X_P$ = -4.54

$Y_P$ = 2.2

MMMMM = 23

LOSC = 4

SUM = 8.93244D-1

SUMA = 9.27960D-1

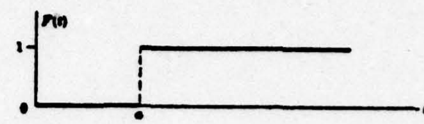
TIME = 1.1341D-1

ANALYTICAL = 8.93244D-1

Case 93

Heaviside's unit function $u(t-a)$

$$\mathcal{L}^{-1}\left[\frac{e^{-as}}{s}\right] =$$




```
      SUBROUTINE VALUE (X,Y,T,AL,U,V,MMMMM)
C     THIS IS TABLE ENTRY NUMBER 139 OF LAPLACE TRANSFORMS BY SPIEGEL
      IMPLICIT REAL*8 (A,B,D-H,O-Z)
      IMPLICIT COMPLEX*16 (C)
      MMMMM = MMMMM + 1
      CS = DCMPLX(X,Y)
      XPTR = X*(T-AL)
      XPTI = Y*(T-AL)
      CXPT = DCMPLX(XPTR,XPTI)
      CRESLT = CDEXP(CXPT)/CS
      U = DREAL(CRESLT)
      V = DIMAG(CRESLT)
      RETURN
      END
```

AL = 1.25D-1

T = 8.0625D+0

AA = 0.2

$X_F$ = -4.64

$y_F$ = 2.2

MMMMM = 23

LOSC = 5

SUM = 1.00000D+0

SUMA = 1.41682D+0

TIME = 6.0554D-2

ANALYTICAL = 1.00000D+0

Case 94

$$\mathcal{L}^{-1}\left[\frac{e^{-a^2/4s}}{s}\right] = J_0\left[at^{1/2}\right]$$

```
      SUBROUTINE VALUE (X,Y,T,AL,U,V,MMMMM)
C     THIS TRANSFORM WAS TAKEN FROM ROBERTS AND KAUFMAN
      IMPLICIT REAL*8 (A-H,O-Z)
      MMMMM = MMMMM + 1
      TEM1 = -1.D+0*AL*AL*X
      TEM2 = AL*AL*Y
      TEM3 = 4.D+0*((X*X) + (Y*Y))
      TEM4 = TEM1/TEM3
      TEM5 = TEM2/TEM3
      TEM6 = DEXP(TEM4)*DCOS(TEM5)
      TEM7 = DEXP(TEM4)*DSIN(TEM5)
      TEM8 = (TEM6*X) + (TEM7*Y)
      TEM9 = (TEM7*X) - (TEM6*Y)
      TEM10 = (X*X) + (Y*Y)
      TEM11 = TEM8/TEM10
      TEM12 = TEM9/TEM10
      TEM13 = DEXP(X*T)*DCOS(Y*T)
      TEM14 = DEXP(X*T)*DSIN(Y*T)
      U = (TEM13*TEM11) - (TEM14*TEM12)
      V = (TEM14*TEM11) + (TEM13*TEM12)
      RETURN
      END
```

AL = 2.5D-1

T = 1.6D+1

AA = 0.2

$X_P$ = -3.04

$y_P$ = 1.8

MMMMM = 19

LOSC = 8

SUM = 7.65198D-1

SUMA = 3.98573D+0

TIME = 6.3648D-2

ANALYTICAL = 7.65198D-1

Case 95

$$\mathcal{L}^{-1}\left[\frac{1}{s}\left[\frac{s-a}{s+a}\right]^2\right] = 1 - 4ate^{-at}$$

```
      SUBROUTINE VALUE (X,Y,T,AL,U,V,MMMMM)
C     THIS TRANSFORM WAS TAKEN FROM ROBERTS AND KAUFMAN
      IMPLICIT REAL*8 (A,B,D-H,O-Z)
      IMPLICIT COMPLEX*16 (C)
      MMMMM = MMMMM + 1
      ZERO = 0.D+0
      CS1 = DCMPLX(X,Y)
      CS2 = DCMPLX(AL,ZERO)
      CS3 = CS1 - CS2
      CS4 = CS1 + CS2
      XPTR = X*T
      XPTI = Y*T
      CXPT = DCMPLX(XPTR,XPTI)
      CNUM = CS3*CS3*CDEXP(CXPT)
      CDENOM = CS1*CS4*CS4
      CRESLT = CNUM/CDENOM
      U = DREAL(CRESLT)
      V = DIMAG(CRESLT)
      RETURN
      END
```

AL = 1.25D-1

T = 8.D+0

AA = 0.2

$x_F$ = -4.64

$y_F$ = 2.2

MMMMM = 23

LOSC = 5

SUM = -4.71518D-1

SUMA = 5.06864D-1

TIME = 6.9186D-2

ANALYTICAL = -4.71518D-1

## BIBLIOGRAPHY


1. Abramowitz, M. and Stegun, I. A., Handbook of Mathematical Functions, Dover Publications Inc., New York, 1972.

2. Alfrey, T., "Non-Homogeneous Stresses in Viscoelastic Media", "Quarterly APPL. MATH.", 2, 113-119, 1944.

3. Bellman, R., Kalaba, R., and Lockett, J., Numerical Inversion of the Laplace Transform, American Elsevier Publishing Co., New York, 1966.

4. Carrier, G. F., Krook, M., and Pearson, C. E., Functions of a Complex Variable, McGraw-Hill Publishing Co., 1966.

5. Doetsch, G., Laplace Transformation, Dover Publications Inc., New York, 1943.

6. Doetsch, G., Handbuch der Laplace Transformation, Verlag, Burkhaeuser, Basle, 1950.

7. Erdelyi, A., Magnus, W., and Tricomi, F. G., Tables of Integral Transforms, Vol II, McGraw-Hill Publishing Co., New York, 1954.

8. Erdelyi, A., "The Inversion of the Laplace Transformation", Mathematical Magazine, Vol. 24, 1950, pp. 1-6.

9. Erdelyi, A., "Note on an Inversion Formula for the Laplace Transformation", Journal London Mathematical Society, Vol. 18, 1943, pp. 72-77.

10. Erdelyi, A., "Inversion Formula for the Laplace Transformation", Philosophical Magazine, Vol. 34, 1943, pp. 533-537.

11. Esch, R. E., "The Instability of a Shear Layer Between Two Parallel Streams", Journal of Fluid Mechanics, Vol. 3, 1957, pp. 289-303.

12. Hiep, D. D., Heat Transfer in Porous Media, Thesis, Naval Postgraduate School, Monterey, California, 1965.

13. McCollum, P. A., and Brown, B. F., Laplace Transform Tables and Theorems, Holt, Reinhart, and Winston, 1965.

14. Roberts, G. E., and Kaufman, H. K., Tables of Laplace Transforms, Saunders, 1966.

15. Salzer, H.E., "Tables of Coefficients for the Numerical Calculation of Laplace Transforms", National Bureau of Standards, Applied Mathematics, Ser. No. 30, U. S. Gov. Print. Office, Washington, 1953.

16. Salzer, H. E., "Tables for the Numerical Caluclation of Inverse Laplace Transforms", Journal of Mathematics and Physics, Vol. 37, 1958, pp. 89-109.

17. Salzer, H. E., "Orthogonal Polynomials Arising in the Numerical Evaluation of Inverse Laplace Transforms", Mathematical Tables and Other Aids to Computation, Vol. 9, 1955, pp. 164-177.

18. Schmittroth, L. A., "Numerical Inversion of Laplace Transforms", ACM Communications, Vol. 3, March 1960.

19. Shanks, D., "Nonlinear Transformations of Divergent and Slowly Convergent Sequences", Journal of Mathematical Physics, Vol. 34, 1955, pp. 1-42.

20. ter Haar, D., "An Easy Approximate Method of Determining the Relaxation Spectrum of a Viscoelastic Material", Journal of Polymer Science, Vol. 6, 247, 1951.

21. Weeks, W. T., "Numerical Inversion of Laplace Transforms Using Laguerre Functions", Journal of the Association of Computational Machinery, Vol. 13, No. 3, 1966, pp. 419-426.

22. Widder, D. V., "An Application of the Laguerre Polynomials", Duke Mathematical Journal, Vol. 1, 1935, pp. 126-136.

23. Zargary, A., Conjugated Transient Heat Transfer in Pipe Flow, Thesis, Naval Postgraduate School, Monterey, California, 1972.

INITIAL DISTRIBUTION LIST

| | | No. Copies |
|---|---|---|
| 1. | Defense Documentation Center<br>Cameron Station<br>Alexandria, Virginia 22314 | 2 |
| 2. | Library, Code 0142<br>Naval Postgraduate School<br>Monterey, California 93940 | 2 |
| 3. | Department Chairman, Code 69<br>Department of Mechanical Engineering<br>Naval Postgraduate School<br>Monterey, California 93940 | 1 |
| 4. | Professor J. E. Brock, Code 69Bc<br>Department of Mechanical Engineering<br>Naval Postgraduate School<br>Monterey, California 93940 | 2 |
| 5. | LCDR John H. Duncan, USN<br>Rt. 4, Box 147<br>Wetumpka, Alabama 36092 | 1 |
| 6. | Dr. William A. Beyer<br>University of California<br>Los Alamos Scientific Laboratory<br>P. O. Box 1663<br>Los Alamos, New Mexico 87544 | 1 |

END